MODELLING OF EMULSION POLYMERIZATION IN A CONTINUOUS STIRRED TANK REACTOR
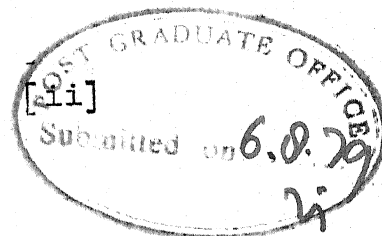
A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of

MASTER OF TECHNOLOGY

By

GIRISH DAMODAR BALLAL

to the

DEPARTMENT OF CHEMICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

AUGUST, 1979

CERTIFICATE

This is to certify that the present work 'MODELLING OF EMULSION POLYMERIZATION IN A CONTINUOUS STIRRED TANK REACTOR' has been carried out under my supervision and has not been submitted elsewhere for a degree.
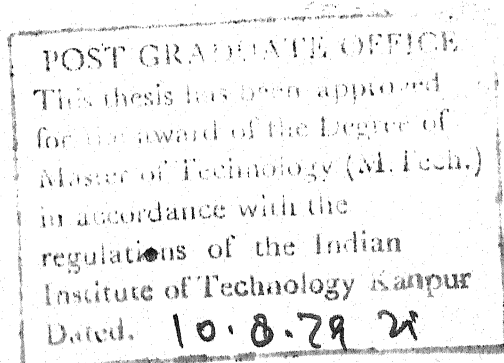
Date: 4th August 1979

[K.S. Gandhi]
Professor
Department of Chemical Engineering
Indian Institute of Technology
Kanpur-208016,India

CHE-1979-M-BAL-MOD

CONTENTS

LIST OF FIGURES

ABSTRACT

A general modelling framework was suggested by Min and Ray [17] for emulsion polymerization reactions. A simplified model based on the framework has been proposed to simulate the emulsion polymerization reaction in a continuous stirred tank reactor. The model uses the population balance equation approach. The method of moments has been used to solve the population balance equations. The model is capable of calculating conversion, rate of polymerization, number of particles, particle size distribution and the molecular weight distribution of resultant polymer latex. The variables studied were residence time and initiator concentration. The results obtained using this model were compared with the experimental data and the theoretical results of DeGraff and Poehlein [21]. The proposed model is found to predict the aforesaid quantities well at low residence times. The predictions of the model are in moderate agreement with data at high residence times. The theoretical approach by DeGraff and Poehlein behaves in the opposite way.

CHAPTER 1

INTRODUCTION

The 50 year old technique of emulsion polymerization
is still being widely used in polymer industry. It often
has significant advantages over homogeneous polymerization.
Its final product is a latex and hence possesses the obvious
advantage of direct application in cases like paints and
coatings. Apart from this, there are distinct advantages
in operating the reactor. The viscosity of a polymer latex
in an emulsion polymerization reactor is low and hence the
temperature control is relatively easy. In addition, a
considerably high molecular weight product can be obtained
at much higher reaction rates. For these reasons, emulsion
polymerization has become commercially important in polymer
industry. At present it is a major processing step in the
manufacture of polymer products such as synthetic rubbers,
paints, inks, coatings, adhesives, high impact strength
co-polymers etc.

The commercial technique of emulsion polymerization
can be divided into two classes: (1) batch and (2) continuous.

Batch systems are relatively simple and were developed
first. These systems, although less efficient, are used
for low production rates and for speciality applications.
The advantages of this technique are the relatively simple
equipment to be handled and the fact that a small amount

of material can be easily processed. This type of poly-
merization has been extensively studied in literature.
Numerous articles are available in literature regarding the
experimental and theoretical aspects of emulsion polymerization
in batch. The more important of these works are by Smith
and Ewart [1], Stockmayer [2], O'Toole [3], Gardon [4-10],
Ugelstad [11], Medvedev [12], Min and Ray [13]. These and
many other works have been comprehensively reviewed by
Alexander and Nappel [14], Vander Hoff[15,16], Min and Ray [17].

As the annual output of the polymeric materials
increases, more and more attention is being given to the
use of continuous reactors. Although continuous stirred
tank reactor (CSTR) is widely used commercially for emulsion
polymerization, very little published information is available
about these reactors. Experimental data is limited since
the equipments are more costly and less flexible, experimental
run times are longer and larger amounts of raw materials
are consumed. Similarly the modelling of these reactors
involves some conceptual problems which are not present in
batch systems.

The earliest of the models developed for emulsion
polymerization in a CSTR is by Greshberg and Longfield [18].
They have employed a residence time distribution model coupled
with the Smith-Ewart kinetics. Omi and co-workers [19]
and Nomura and co-workers [20] have employed Smith-Ewart
kinetics for different reaction systems. They have also used

different models for initiation of particles. DeGraff
and Poehlein [21] have employed Stockmayer's [2] theory
coupled with residence time distribution for modelling a
single stirred tank reactor. Recently population balance
approach has been employed for modelling of continuous
reactors.

Stevens and Funderburk [22] have employed a simple
population balance approach to model a CSTR. Further work
using this approach was done by Thompson and Stevens [23],
Thompson and co-workers [24] and Cauley and Thompson[25].
These authors have used population balance equations to
generate radical number distribution. They have used size
independent particle growth rate. Cauley and Thompson[25]
have further assumed instantaneous termination of radicals
inside the particles. They have not developed material
balance equations for various reactive species. The conversion
and molecular weights of the polymer could not be calculated.

Min and Ray [17] have developed a comprehensive
model for emulsion polymerization in batch and CSTR. They
have taken into consideration a number of mechanisms like
desorption of radicals from particles, chain transfer,
initiation in aqueous phase, end-group stabilization of
polymer particles, initiation by precipitation etc. They
have, however, not presented any numerical result from
this comprehensive model for CSTR.

The latest work on modelling of CSTR is by Kirilov
and Ray [26]. They have used the general model of Min and
Ray [17] for a specific case by making a number of simplifi-
cations. They have employed simple population balance
equation for particle size distribution and have used
Stockmayer's [2] relation for calculating the average number
of radicals per particle. They have calculated average
particle size and conversion obtained in the reactor. They
did not calculate either the particle size distribution or
the molecular weights of the polymer product.

In the present work, an attempt has been made to develop
and solve modelling equations for emulsion polymerization
in a CSTR. The population balance approach has been used.
The population balance equations employed are more comprehen-
sive. As a result, the molecular weight distribution of
the resultant product can also be generated in addition to
conversion and the particle size distribution.

CHAPTER 2

PHYSICAL MODEL AND POPULATION BALANCE EQUATIONS

Before starting the actual formulation of modelling
equations, it is necessary to understand the physical picture
and details of various mechanisms operating in an emulsion
polymerization reactor. The currently accepted [14-17]
picture is as follows:

A typical emulsion polymerization recipe consists of
100 parts of water, 50 parts of monomer, 2 parts of emulsifier,
0.1 part of initiator and small amounts of chain transfer
agents and other additives. The soap is present in micelles,
on the surface of monomer droplets and in the aqueous phase.
The monomer is present mainly as droplets and a small amount
(1 per cent) is solubilized in the micellar soap. A small
portion of the monomer may also be dissolved in the aqueous
phase. Generally, the initiator is present in the aqueous
phase. In early stages of the reaction, the system consists
of 3 types of particles dispersed in continuous aqueous
phase: (1) emulsified monomer droplets of size 10000 $\overset{o}{A}$ and
at a concentration of 10^{12}/ml, (2) soap micelles with size 50-
100 $\overset{o}{A}$ and at a concentration of 10^{18}/ml and (3) emulsifier
stabilized monomer swollen polymer particles of size 500-
1000 $\overset{o}{A}$, and at a concentration of 10^{14}-10^{15}/ml in later
stages.

As the reaction proceeds the polymer particles grow in size, consuming the monomer present in the whole system, thus requiring more and more amount of soap to stabilize them. The soap is supplied by micelles which decrease in number and finally disappear. Similarly, the monomer droplets shrink in size as they supply monomer to the polymer particles and finally disappear, leaving the reaction system consisting of polymer particles dispersed in the aqueous phase.

Now some of the mechanisms operating in the reaction mixture will be discussed.

Nucleation of polymer particles is one of the least understood phenomena. It can occur in two different ways. The initiator present in the aqueous phase thermally decomposes into constituent radicals (reaction temperature approximately $50-70^{\circ}C$) and these radicals grow by adding the monomer units from the monomer dissolved in the aqueous phase. During this process the radicals and the active polymer chains exhibit Brownian motion. During their motion they can collide with micelles, get absorbed into them and start the polymerization reaction thereby converting the micelles into polymer particles. They can also collide with and get absorbed in existing polymer particles. The radicals which are not captured by micelles or particles continue to add on monomer units and in the process decrease their

aqueous phase solubility. When a critical limit of chain
size is reached they will precipitate and initiate new particles.
Both these mechanisms of particle nucleation operate simul-
taneously. The former is more important for monomer scarcely
soluble in water, while the latter is predominant for
monomers which are relatively more soluble in water.

The particles thus initiated consist of monomer,
polymer chains and the live polymer radicals. Polymerization
continues in the particles, the monomer being supplied by
monomer droplets. Radicals absorb into and desorb from the
particles. Propagation, termination as well as chain transfer
can take place inside particles. Soap required for stabili-
zation is supplied by the micelles until they disappear.
As more and more amount of monomer is consumed the monomer
droplets decrease in size and number. Ultimately they
disappear leaving only the polymer particles containing live
radicals and dissolved monomer. As the reaction proceeds to
higher range of conversion the monomer concentration in a
particle decreases, viscosity of reaction medium increases and
the reaction can become diffusion controlled. Gel effect
may play an important role at higher conversion. Apart from
these mechanisms, particles can also coagulate with each
other to a limited extent. The extent of coagulation depends
upon the particle size, amount of soap adsorbed on each
particle, reaction temperature etc.

The polymer particles consist of monomer, polymer chains and live radicals as stated earlier. However, the exact structure of these particles is not yet known. The monomer and polymer may solubilize each other to form a homogeneous mixture. If that is the case, reaction will occur at a constant rate throughout the homogeneous particle. On the other hand, Williams [27] has postulated a non-uniform morphology for the particles, comprising of a polymer rich core and a monomer rich shell. The reaction is assumed to take place on the growing monomer polymer interface. Naper[28] has suggested exactly the opposite structure consisting of a monomer with core and a polymer rich shell.

This, in short, is a physical picture of the emulsion polymerization process. The aforesaid mechanisms operate simultaneously in a batch or continuous emulsion polymerization reactor. The objective of the present work is to use the knowledge of these basic mechanisms to model the character- istics of a polymer latex produced in a continuous stirred tank reactor.

The problem can be approached in two different ways:

In the residence time distribution approach the segregated flow model of a CSTR is employed. In this model the reactor is assumed to consist of a number of small independent reaction units. Each reaction unit is assumed to reside for different times inside the reactor. The reaction

units are assumed to operate as isolated batch reactors.
The residence time distribution of different units combined
with the knowledge of batch behaviour is believed to represent
the CSTR characteristics. This method was most recently
used by DeGraff and Poehlein [21] for the modelling of
emulsion polymerization CSTR. Here a particle was identified
as a reaction unit. They assumed the usual exponential
residence time distribution. Another approach for modelling
is the use of population balance equations which will be
employed in this work. A short discussion about the theory
of population balance equations is available in ref. 29.
A brief introduction to this method is given below.

For any particulate system, a particle phase space
consisting of a minimum number, say p, of independent co-
ordinates, attached to a particle distribution, which
completely describes the properties of the distribution, is
postulated. The particle phase space may be divided into
two subregions consisting of m internal and three external
co-ordinates. External co-ordinates refer to the three
spatial co-ordinates of the particle. Internal co-ordinates
quantitatively represent the state of the particle independent
of its spatial position.

Let us consider a (m+3) dimensional particle distri-
bution function n(Z,t) defined over region Z consisting of
three external co-ordinates and m independent internal

co-ordinates. In a dynamic particulate system, individual particles change their position in the particle phase space. If the change of particle co-ordinates is gradual and continuous in any direction the rate of such 'motion' is called as convective particle velocity along that co-ordinate axis. Thus the particle phase-space velocity vector is defined as

$$\vec{v} = v_x\,\delta_x + v_y\delta_y + \dots v_m\delta_m + v_1\delta_1 + v_2\delta_2 + v_3\delta_3 \tag{2.1}$$

$$= \vec{v}_{int} + \vec{v}_{ext}$$

where v's are the components of the particle velocity and δ's are the unit vectors along the various co-ordinate axes. In general, the external velocity $\vec{v}_{ext}$ can be calculated from the fluid velocity and momentum exchange relationship between fluid and particle. The internal particle velocity $\vec{v}_{int}$ is assumed to be unique function of state of the particle.

There is also a possibility of particles suddenly appearing or disappearing in particle phase space along any co-ordinate. This possibility is represented by the birth and death functions of particle distribution. In general, these birth and death functions can be related to the state of the total system and the position of a particle in the particle phase space.

The population balance equation for particles in some fixed subregion Z_1 of particle phase space will be based on

the usual conservation law.

Accumulation = Input − Output + net generation

$$(2.2)$$

Consider the subregion to move convectively with particle phase space velocity $\vec{v}$. The population balance equation can be written as

$$\frac{d}{dt} \oint_{Z_1} n \, dZ = \int_{Z_1} (B-D) \, dZ \qquad (2.3)$$

where B and D represent birth and death functions. Expanding L.H.S.

$$\frac{d}{dt} \int_{Z_1} n \, dZ = \int \frac{\partial n}{\partial t} \, dZ + n \frac{d\vec{x}}{dt} \Big|_{Z_1}$$

$$= \int_{Z_1} [\frac{\partial n}{\partial t} + \vec{\nabla} \cdot (\frac{d\vec{x}}{dt} n)] \, dZ$$

where $\vec{x}$ are the co-ordinates of system.

Replacing $\frac{d\vec{x}}{dt}$ by $(\vec{v}_{ext} + \vec{v}_{int})$ and combining with 2.3 gives the following expression:

$$\int_{Z} [\frac{\partial n}{\partial t} + \vec{\nabla} \cdot (\vec{v}_{ext} \cdot n) + \vec{\nabla} \cdot (\vec{v}_{int} \cdot n) + D-B] \, dZ = 0$$

The equation is valid for any arbitrary region Z_1. Hence the integrand vanishes for all Z.

Therefore $\frac{\partial n}{\partial t} + \vec{\nabla} \cdot (\vec{v}_{ext} \cdot n) + \vec{\nabla} \cdot (\vec{v}_{int} \cdot n) + D-B = 0$

$$(2.4)$$

This is a general expression which can be used to describe any particulate system, provided the number of

particles in the system is sufficiently large so that
$n(Z,t)$ can retain its statistical significance.

This equation will now be applied to a continuous
well stirred particulate system of volume $V(t)$, having a
population density function $n(Z,t)$, p input and q output
streams. Since in a well stirred system, tracing of external
coordinates of particles is not of interest, this equation
can be integrated over the range of external co-ordinates V or
volume occupied by the system

$$\int_V (\frac{\partial n}{\partial t} + \vec{V} \cdot \vec{v}_{int} + D-B + \vec{\nabla} \cdot \vec{v}_{ext} \cdot n) \, dV = 0$$

Therefore $V[\frac{\partial n}{\partial t} + \vec{V} \cdot \vec{v}_{int} \cdot n + D-B] + \int_V \vec{\nabla} \cdot \vec{v}_{ext} \cdot n \, dV = 0$

$$(2.5)$$

The integral can be transformed into a surface
integral of population flux over the surfaces of the system.
The surfaces are: the cross-sections of input-output streams,
free interface of liquid and total fluid-particle interface.
The final expression is

$$\int \vec{\nabla} \vec{v}_{ext} \cdot n \, dV = n \frac{dV}{dt} - \sum_p Q_i n_i + \sum_q Q_o n \qquad (2.6)$$

where Q's are the flow rates of input-output streams.
Combining 2.5 and 2.6 and simplifying →

$$\frac{\partial n}{\partial t} + \vec{V} \cdot \vec{v}_{int} \cdot n + n \frac{d(\log V)}{dt} = B-D + \sum_p \frac{Q_i n_i}{V} - \sum_q \frac{Qn}{V}$$

$$(2.7)$$

This expression represents the macroscopic population balance equation for a particulate, well mixed system. With the knowledge of input-output streams, expressions for velocity $\vec{v_i}$, birth and death functions this equation can be used to model emulsion polymerization in a CSTR mathematically. The values of population density $n(t)$ can be obtained and used to calculate the characteristics of the resultant polymer latex.

In next chapter, this equation will be used to develop the design equations for an emulsion polymerization CSTR.

CHAPTER 3

FORMULATION OF MODELLING EQUATIONS

From the previous chapters it can be seen that an emulsion polymerization latex is a particulate system comprising of a continuous aqueous phase and a particulate phase consisting of polymer particles, micelles and monomer droplets. A set of equations, consisting of material balance equations for continuous phase and population balance equations for particulate phase, completes the mathematical representation of physical polymerization system. Energy balance equations for the system are generally unnecessary because of low latex viscosity and high heat transfer rates, which result in an isothermal operation.

In the following pages development of the modelling equations will be given. This presentation closely follows the approach given by Min and Ray [17]. For convenience, nomenclature for this chapter is provided at the end of the chapter.

Kinetic Scheme: The usual kinetic scheme involving initiation, propagation, termination as well as chain transfer to monomer and transfer agents has been used. The scheme is given below:

1. **Initiation:**

$$I_w \xrightarrow{k_d} R+R = P_1+P_1$$

$$I_p \xrightarrow{ki_p} R+R = P_1+P_1$$

2. <u>Propagation</u>: $P_n + M \xrightarrow{k_p} P_{n+1}$

3. <u>Termination</u>: $P_n + P_m \xrightarrow{k_{tc}} M_{n+m}$

$P_n + P_m \xrightarrow{k_{td}} M_n + M_m$

4. <u>Chain Transfer</u>: $Tr_p + P_n \xrightarrow{k_{ft}} M_n + P_1$

$M_p + P_n \longrightarrow M_n + P_1$
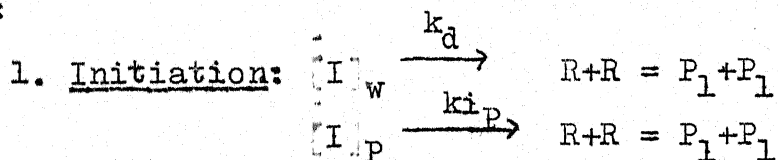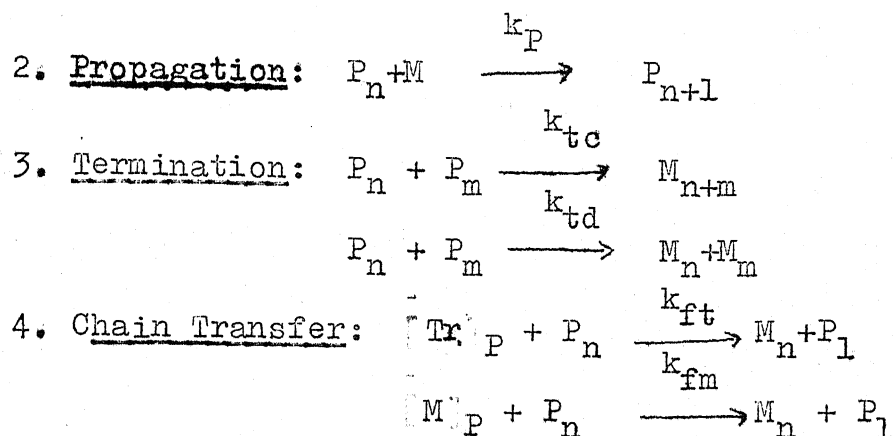
Particulate Phase:

For characterizing a polymer latex, the polymer particles are of prime importance. Very little, if any, reaction takes place in the monomer droplets or micelles. Hence it is not necessary to develop separate population balance equations for these species. It will be assumed that these distributions can be represented by average micelle size (volume) v_m and average droplet size (volume) v_d respectively.

With the polymer recipe and average residence time of the reactor usually employed in practice, the number density of polymer particles is of the order of $10^{14}/cm^3$ emulsion. Hence we can attach an appropriate distribution function to the particles and use macroscopic population balance equation as given in the last chapter.

$$\frac{\partial n}{\partial t} + \vec{\nabla} \cdot \vec{v}_{int} \cdot n + n \frac{d}{dt}(\log V) = B - D + \sum_p \frac{Q_i n_i}{V} - \sum_q \frac{Q n}{V}$$

$$(2.7)$$

Consider a reactor of volume V, with an input and output stream, both having a volumetric flow rate of q. The

The contents of reactor are assumed to be completely
mixed. Let r_i represent the net generation term (B-D)
resulting in discrete changes of variables due to the
various mechanisms. Let n be the distribution function
under consideration. The aforesaid equation, when applied
to such system becomes

$$\frac{\partial n}{\partial t} + \vec{\nabla} \cdot \vec{v}_{int} \cdot n = \sum_{i=1}^{m} r_i + \frac{1}{\Theta} n \Big|_{feed} - \frac{1}{\Theta} n \qquad (3.1)$$

where Θ represents the mean residence time and m
depends upon the number of mechanisms to be included in the
model.

Radical Number Distribution:

Consider a distribution function $f(i,v,t)$ which
represents number of particles per cm^3 of emulsion, having
volume v to v+dv and i radicals, at any time t. The knowledge
of this distribution is necessary to calculate the rate of
polymerization and $\bar{i}_{avg}$. The population balance equation
corresponding to this function is

$$\frac{\partial f(i,v,t)}{\partial t} + \frac{\partial}{\partial v} \cdot (r_v \cdot f(i,v,t)) = \sum_{i=1}^{m} r_i + \frac{1}{\Theta} f(i,v,t) \Big|_{feed}$$
$$- \frac{1}{\Theta} f(i,v,t) \qquad (3.2)$$

First term represents rate of change of $f(i,v,t)$ with
respect to time. r_v in the second term represents the rate
of change of volume of particle with respect to time, [ie $\frac{d\bar{V}}{dt}$].

The size of a particle varies due to the propagation reaction taking place inside the particle and due to particle swelling. This quantity will depend upon the model assumed for particle structure. For homogeneous particle morphology Garden [4] has calculated this quantity by correlating the propagation mechanism with the monomer concentration and polymer and monomer densities. He obtained following relationship for the conditions when the monomer droplets are present in the system and the aqueous phase is saturated (with monomer).

$$\frac{dv}{dt} = r_v = \frac{k_P}{N_A} \cdot \frac{d_m}{d_P} \cdot \frac{\bar{\varphi}}{(1-\bar{\varphi})} \cdot i \qquad (3.3)$$

The rate terms r_i in the RHS of 3.2 depend upon the particular mechanisms to be taken into consideration. These terms will be discussed below.

First term, r_1 represents the change in f (i,v,t) due to absorption of radicals into the particles. Radicals can either propagate and precipitate as oligomers in the aqueous phase or can be absorbed into the particles. The relative importance of these two mechanisms depends upon the monomer solubility and the propagation rate constant in the aqueous phase. Gardon [4] has obtained an expression for rate of radical absorption by computing the probability of collision of a radical with micelles and particles. The final expression is

$$R_{ia} = \frac{2 f k_d [I]_w}{A_m + A_p} \qquad \text{when} \quad S \gg 4/L$$

$$= \frac{2 f k_d [I]_w}{(4/L)} \qquad \text{when} \quad S \ll 4/L$$

where L represents the average length travelled by a growing chain before it reaches the critical length at which it precipitates and is given by

$$L = (2D\ t_{prec})^{0.5}$$

Here t_{prec} represents the time required by radical to reach critical chain length for precipitation and is a function of the propagation rate constant and monomer concentration in the aqueous phase. D is the diffusion coefficient of radicals in the aqueous phase.

Rate r_1 in eqn.3.2 is then given by the following expression

$$r_1 = R_{ia} \times a_p\ N_A\ (f(i-1,v,t) - f(i,v,t)) \qquad (3.4)$$

where the first term represents the formation of $f(i,v,t)$ by absorption of a radical into $f(i-1,v,t)$ type of particles and the second term represents destruction of $f(i,v,t)$ type of particle by absorption of a radical, thus transforming it into a $f(i+1,v,t)$ type particle.

Second rate term (r_2) represents the effect of formation of a radical pair inside the particle by the decomposition of monomer-soluble initiator.

$$r_2 = f\, k_{ip}[I]_P \cdot N_A \cdot v\, (f(i+2,v,t) - f(i,v,t)) \quad (3.5)$$

Radicals can also desorb from the particle. This effect can be accounted for by the following expression:

$$r_3 = \frac{k_o a_P}{v} [(i+1)\, f(i+1,v,t) - i\, f(i,v,t)] \quad (3.6)$$

First term represents the formation of $f(i,v,t)$ type particle by radical desorption from a $f(i+1,v,t)$ type particle and second term represents the destruction of $f(i,v,t)$ by desorption of a radical.

Rate of change of $f(i,v,t)$ by termination of two radicals inside the particle, is given by the following expression:

$$r_4 = \frac{k_{tc} + k_{td}}{2} \left(\frac{1}{N_A v}\right) [(i+2)(i+1)\, f(i+2,v,t) - (i)(i-1)f(i,v,t)]$$

$$(3.7)$$

The rate of formation of new particles by radical absorption in micelles is given by

$$r_5 = R_i \times \frac{a_m \cdot m}{A_m + A_P}\, \delta_{i1}\, \delta(v - v_m) \quad (3.8)$$

δ functions on right hand side of the expression indicate that new particles, thus formed, will have volume v_m and will contain one radical.

Particle Size Distribution:

Let $F(v,t)$ be the particle size distribution function. This distribution is quite important since the particle size is one of the characterizing properties of a polymer latex.

Consider the balance equation for $F(v,t)$, which represents the number of particles having volume between v and $v+dv$ at any time t. It is related to radical number distribution by the following equation:

$$F(v,t) = \sum_{i=0}^{\infty} f(i,v,t) \tag{3.9}$$

By combining 3.2 to 3.9 the population balance equation for $F(v,t)$ can be derived to be

$$\frac{\partial F(v,t)}{\partial t} \qquad + \frac{\partial}{\partial v}(r_v \, F(v,t)) \qquad =$$

Rate of change of   Rate of change of $F(v,t)$ due to
$F(v,t)$ with respect   + volume growth of particles
to time

$$R_i \, \frac{A_m}{A_m + A_p} \cdot \delta(v - v_m) \qquad + \frac{1}{\theta} [F(v,t) \big|_{feed} - F(v,t)]$$

Rate of change of $F(v,t)$ Change of $F(v,t)$ due to
due to generation of new   + input and output
particles of size v_m

$$\tag{3.10}$$

where

$$r_v = \frac{k_p}{N_A} \cdot \frac{d_m}{d_p} \cdot \frac{\bar{\varphi}}{(1-\bar{\varphi})} \cdot \bar{i}_{avg} \tag{3.11}$$

and $\bar{i}_{avg}$ [i.e. average number of radicals per particle] is given by

$$\bar{i}_{avg} = \frac{\int_{v_m}^{\infty} \sum_{i=0}^{\infty} i \, f(i,v,t) \, dV}{\int_{v_m}^{\infty} \sum_{i=0}^{\infty} f(i,v,t) \, dv} \tag{3.12}$$

Live Polymer Chain Length Distribution:

In order to calculate the molecular weight distribution the chain length distribution of the polymer radicals present inside the particle is needed. This distribution is used to solve the dead polymer chain length distribution which is in turn used to obtain the m.w.d. of the polymer product. Consider the particles having volume between v and v+dv and i radicals at any time t. These i radicals will in general have different chain lengths. Amongst these radicals consider the once having chain length n. Let $\bar{L}(n,i,v,t)$ represent the total number of radicals of chain length n present in all the particles having volume between v and v+dv and having i radicals, at any time t. The population balance for such radicals is given by the following expression.

$$\frac{\partial\, L(n,i,v,t)}{\partial t} + \frac{\partial}{\partial v}(r_v \cdot \bar{L}(n,i,v,t)) = \sum_{i=1}^{k} r_i$$

$$+ \frac{1}{\theta}\left[\bar{L}(n,i,v,t)\Big|_{feed} - L(n,i,v,t)\right]$$

$$(3.13)$$

First term represents change in $L(n,i,v,t)$ with respect to time. The second term represents change in $L(n,i,v,t)$ due to change in size of a particle, where r_v represents the rate of change of volume with respect to time.

The rate terms r_i on right hand side, as before, depend on specific assumptions made. Firstly consider the effect of absorption of radicals. It has been assumed that

only primary radicals can absorb into the particle and that the particle nucleation by precipitation in aqueous phase has been neglegible. Let $g(n,i,v,t)$ represent number of radicals of chain length n in a $f(i,v,t)$ type particle. When a radicalis absorbed in a $f(i-1,v,t)$ type particle it converts the particle into a $f(i,v,t)$ type, making in the process, a contribution of $g(n,i-1,v,t)$ radicals to $L(n,i,v,t)$. Similarly when a primary radical is absorbed in a $f(i,v,t)$ type particle $g(n,i,v,t)$ number of radicals will be lost from $L(n,i,v,t)$. When a primary radical is absorbed in a particle it adds only to $L(1,i,v,t)$. Thus the total effect, per cm^3 of emulsion can be represented as:

$$r_1 = R_i \; \frac{a_p}{A_p + A_m} \cdot f(i-1,v,t) \cdot g(n,i-1,v,t)$$

$$-R_i \; \frac{A_p}{A_p+A_m} \; f(i,v,t) \cdot g(n,i,v,t)$$

$$+R_i \; \frac{A_p}{A_p+A_m} \; f(i-1,v,t) \; \delta n_1$$

$$= R_i \; \left[\frac{a_p}{A_p+A_m}\right]\left[f(n,i-1,v,t) - f(n,i,v,t) + f(i-1,v,t)\delta n_1\right]$$

$$(3.14)$$

From the definition

$$g(n,i,v,t) = \frac{L(n,i,v,t)}{f(i,v,t)} \tag{3.15}$$

The initiator dissolved in a polymer particle can decompose to give a pair of radicals of chain length one each. A contribution of $g(n,1-2,v,t)$ is made to $L(n,i,v,t)$

per generation of a pair due to such an event. In addition
to this the radicals formed themselves contribute to
addition of radicals with chain length one. The same process
occurring in a f(i,v,t) type particle would result in a
loss to L(n,i,v,t)

$$r_2 = k_{ip}[I]_P \ v \cdot N_A \ [(g(n,i-2,v,t) + 2\delta n_1)f(i-2,v,t)$$

$$-g(n,i,v,t) \cdot f(i,v,t)]$$

$$= k_{ip} [I]_P \cdot v \cdot N_A[L(n,i-2,v,t) - L(n,i,v,t) +2\delta n_1 f(i-2,v,t)]$$

$$(3.16)$$

The effect of radical desorption from particles is
given by

$$r_3 = k_o \ \frac{a_P}{v}[\ (i+1)f(i+1,v,t) \cdot g(n,i+1,v,t)(1- \frac{1}{i+1})$$

$$- i \ f(i,v,t) \ g(n,i,v,t)]$$

$$= \frac{k_o \ a_P}{v} \ [\ (i) \ L(n,i+1,v,t) - (i) \ L(n,i,v,t)] \quad (3.17)$$

Here the first term represents the production of L(n,i,v,t)
by desorption of radicals from f(i+1,v,t) except when a
radical of length n desorbs. The second term represents loss
due to desorption from radicals from f(i,v,t)type particles.

The effect of the kinetic steps inside the polymer
particle depends upon the structure of particle. Assuming
homogeneous particle morphology, the effect of propogation
and transfer reaction can be expressed as

$$r_4 = k_p[M]_p\, f(i,v,t)\cdot g(n-1,i,v,t) - k_p[M]_p\, f(i,v,t)\, g(n,i,v,t)$$

$$- k_{ft}[Tr]_p\, f(i,v,t)\, g(n,i,v,t) + k_{ft}[Tr]_p\cdot i\, f(i,v,t)\,\delta n_1$$

$$-k_{fm}[M]_p\, f(i,v,t)\, g(n,i,v,t) + k_{fm}[M]_p\, i\, f(i,v,t)\,\delta n_1$$

$$= k_p[M]_p[L(n-1,i,v,t) - L(n,i,v,t)] + k_{ft}[Tr]_p[-L(n,i,v,t)$$

$$+f(i,v,t)\cdot i\cdot \delta n_1] + k_{fm}[M]_p[f(i,v,t)\cdot i\cdot \delta n_1 - L(n,i,v,t)]$$

$$(3.18)$$

The first two terms represent the effect of propagation;
third and fourth term represent chain transfer to transfer
agent and the fifth and sixth term represent chain transfer
to monomer. The chain transfer reactions involve addition
of one chain to $f(1,i,v,t)$. This effect has been accounted
for in fourth and sixth term.

The effect of termination reactions can be written as

$$r_5 = \frac{k_{tc}+k_{td}}{N_A v}\left[\left(\frac{(i+2)(i+1)}{2} - (i+1)\right)\cdot L(n,i+2,v,t)\right.$$

$$\left. - \frac{(i)(i-1)}{2}\, L(n,i,v,t)\right]$$

$$= \frac{k_{tc}+k_{td}}{N_A v}\left[\frac{(i+1)(i)}{2}\, L(n,i+2,v,t) - \frac{(i)(i-1)}{2}\, L(n,i,v,t)\right]$$

$$(3.19)$$

First and third terms in the initial expression
represent change due to termination between any two radicals.
Third term represents the termination between a radical of
chain length n and remaining $(i+1)$ radicals. The possibility

of termination between two radicals of same chain length n, which will be a second order term, has been neglected.

The effect due to radical absorption in micelles is given by the following expression:

$$r_6 = R_i \frac{a_m}{(A_p + A_m)} \cdot m \cdot l \cdot \delta n_1 \, \delta(v - v_m) \delta_{il}$$

$$= R_i \frac{A_m}{(A_p + A_m)} \, \delta n_1 \, \delta(v - v_m) \, \delta_{il} \qquad (3.20)$$

Dead Polymer Chain Length Distribution:

This distribution is related to the chain length of dead polymer molecules inside the particles and is thus closely connected with m.w.d. of the resultant polymer latex. Consider the particles having volume between v and v+dv and possessing i active radicals at any time t. Apart from these i active radicals, each particle contains some dead polymer chains. Let G(n,i,v,t) represent total number of these dead chains contained in all f(i,v,t) type particles at any time t. The population balance equation for this function can be written as

$$\frac{\partial G(n,i,v,t)}{\partial t} + \frac{\partial \, r_v \cdot G(n,i,v,t)}{\partial v} = \sum_{i=1}^{k} r_i$$

$$+ \frac{1}{\theta} [G(n,i,v,t) \Big|_{feed} - G(n,i,v,t)] \qquad (3.21)$$

First term represents the change in G(n,i,v,t) with respect to time and the second term expresses the change due to change in volume of particle.

Rate terms on the right hand side depend upon the particular mechanisms to be considered in the model. Derivation of these terms is similar to that for corresponding terms in live polymer chain length distribution. These terms will be discussed briefly.

The first rate term r_1 represents the effect of radical absorption on $G(n,i,v,t)$. Lt $g'(n,i,v,t)$ represent number of dead polymer chains per particle. Hence $G(n,i,v,t) = g'(n,i,v,t) f(i,v,t)$. When a new radical enters a polymer particle of type $f(i-1,v,t)$, it converts that particle into $f(i,v,t)$ type particle and a contribution of $g'(n,i-1,v,t)$ is made towards $G(n,i,v,t)$. Hence the total contribution of this mechanism towards change in $G(n,i,v,t)$ is given by

$$r_1 = R_i \frac{a_p}{(A_m+A_p)} f(i-1,v,t) \ g'(n,i-1,v,t) - R_i \frac{a_p}{(A_m+A_p)}$$

$$f(i,v,t) \ g'(n,i,v,t)$$

$$= R_i \frac{a_p}{(A_m+A_p)} [G(n-1,i,v,t) - G(n,i,v,t)] \qquad (3.22)$$

Similarly the effect of radical initiation in particles and desorption from particles can be shown to be given by following expression. It is assumed here that dead polymer chains can not desorb from particles.

$$r_2 = f_{kd} \cdot [I]_p [G(n,i-2,v,t) - G(n,i,v,t)] \qquad (3.23)$$

$$r_3 = \frac{k_o a_p}{v} [(i+1) \ G(n,i+1,v,t) - i \ G(n,i,v,t)] \qquad (3.24)$$

The effect of the kinetic steps is more important in case of $G(n,i,v,t)$ and can be given by following expression:

$$r_4 = k_{ft}[Tr]_P \, L(n,i,v,t) + k_{fm}[M]_P \, L(n,i,v,t)$$
$$+ \frac{k_{tc}+k_{td}}{2} \, [(i+2)(i+1) \, G(n,i+2,v,t) - (i)(i-1) \, G(n,i,v,t)]$$
$$+ \frac{k_{tc}}{2v} \, [f(i+2,v,t) \sum_{m=1}^{n-1} g(m,i+2,v,t) \, g(n-m,i+2,v,t)]$$
$$+ \frac{k_{td}}{2v} \, [g(n,i+2,v,t) \, f(i+2,v,t)] \qquad (3.25)$$

In this expression the first two terms represent formation of dead polymer chains due to chain transfer with transfer agent and with monomer. Third term expresses the change in $G(n,i,v,t)$ by termination of any two radicals in $f(i+2,v,t)$ type particle (i.e. formation) and in $f(i,v,t)$ type particle (destruction). Fourth term represents the additional contribution when two terminating radicals form a polymer chain of length n. Last term represent the possibility of disproportionation involving a live polymer radical of chain length n.

Apart from these population balance equations, balances which define the environment of particles must also be written. These equations include material balances for various reactive species and balances for number of droplets, micelles etc.

Emulsifier Balances:

The total amount of soap present per cm^3 emulsion is given by following expression:

$$\frac{d(S)_T}{dt} = \frac{1}{\theta} [[S]_T \Big|_{feed} - [S]_T]$$ (3.26)

This soap may only be present in aqueous phase as micelles or may be absorbed on the particles and droplets. The process of transfer of emulsifier will be assumed to be extremely fast. Under these conditions the balance for number of micelles yields

$$\frac{m \, a_m}{C E} = [S]_T - S_{wc} - \frac{A_P}{CE} - \frac{m_d \, v_d}{CE}$$ (3.27)

The terms on right hand side represent total soap, the soap present in the aqueous phase, the soap adsorbed on particles and the soap adsorbed on the monomer droplet respectively.

Total Monomer Balance:

The balance for the total amount of monomer present in the reactor is given below:

$$\frac{d \, V_R[M]_T}{dt} = \frac{V_R}{\theta} [[M]_T \Big|_{feed} - [M]_T]$$

$$- V_R \, k_p[M]_P \int_0^\infty \sum_{i=0}^\infty i \, f(i,v,t) \, dv$$ (3.28)

The first two terms represent the effect due to input and output terms. The third term represents the loss of monomer due to propagation in particles. Here the small contributions due to chain transfer and propagation in aqueous phase are neglected.

Monomer Droplets Balance:

Monomer droplets balance can be written as follows. As in the previous case the dynamic of transfer of monomers will be assumed to be extremely fast.

$$m_d v_d = [M]_T V_{mol} - \int_0^\infty v[M]_P V_{mol} \cdot F(v) \cdot dv$$

$$- V_{mol} \; m \cdot [M]_m V_{mol} \qquad (3.29)$$

The terms on right hand side of equation represent total monomer content, monomer in particles, and monomer in micelles.

Aqueous Phase Balances:

The balance for concentration of the initiator in aqueous phase is given below

$$\frac{d \; V_w[I]_w}{dt} = - V_w \; k_d[I]_w + \frac{V_w}{\theta} [[I]_{w \; feed} - [I]_w] \qquad (3.30)$$

Lastly an expression for the monomer concentration in polymer (i.e. $[M]_P$ or $\bar{\varphi}$) is required. The nature of the equation will, in general, depend upon the model for particle structure considered. For a homogeneous particle morphology the free energy of mixing of polymer can be equated to the surface free energy [4]. The resultant expression for $\bar{\varphi}$ is:

$$(1-\bar{\varphi}) + \ln(\bar{\varphi}) + \chi (1-\bar{\varphi})^2 = \frac{2 \gamma V_{mol}}{RT \; r} \qquad (3.31)$$

So far the population balance equations for particulate phase and the material balance for aqueous phase have been

discussed. These balances are assumed to represent the given system completely. These equations can be applied to a practical system and the results can be used to obtain desired characteristics of the resultant polymer latex.

NOMENCLATURE

CE surface coverage by the emulsifier, cm^2/mole

a_m surface area of micelles, cm^2

a_p surface area of particle of size v, cm^2

A_m total surface area of micelles, cm^2/cm^3 emulsion

A_p total surface area of particles, cm^2/cm^3 emulsion

d_p density of polymer, gms/cm^3

d_m density of monomer, gms/cm^3

f efficiency of initiator decomposition, dimensionless

$f(i,v,t)$ number of particles containing i radicals and possessing
volume between v to v+dv at time t, moles/cm^3 emulsion

$F(v,t)$ number of particles having volume between v and v+dv
at time t, moles/cm^3 emulsion

$g(n,i,v,t)$ number of live chains of chain length n present
in a particle containing i radicals and possessing
volume between v to v+dv at time t, dimensionless

$g'(n,i,v,t)$ number of dead chains of chain length n present in
particle containing i radicals and possessing volume
between v to v+dv at time t, dimensionless

$G(n,i,v,t)$ number of dead chains of chain length n present
in all the particles having i radicals and possessing
volume v to v+dv, at time t, moles/cm^3 emulsion

i number of radicals in a particle, dimensionless

$\bar{i}_{avg}$ average number of radicals per particle, dimensionless

$[I]_w$ initiator concentration in aqueous phase, moles/cm^3, water

$[I]_p$ initiator concentration in polymer particles, moles/cm^3

k_{ip} rate constant for dissociation of initiator in polymer phase, 1/cms

k_d rate constant for dissociation of initiator in aqueous phase, 1/cms

k_p rate constant for propagation, cm^3/mole. sec.

k_{tc} rate constant for termination by combination, cm^3/mole.sec.

k_{td} rate constant for termination by disproportionation, cm^3/mole sec

k_{ft} rate constant for chain transfer to transfer agent, cm^3/mole sec.

f_{fm} rate constant for chain transfer to monomer, cm^3/mole sec.

k_o rate constant for radical desorption, cms/sec

$L(n,i,v,t)$ number of live chains of chain length n in all the particles having i radicals and possessing volume between v and v+dv at time, t, moles/cm^3 emulsion

m number of micelles, moles/cm^3 emulsion

$[M]_p$ concentration of monomer in the particles, moles/cm^3

$[M]_m$ concentration of monomer in the micelles, moles/cm^3

$[M]_T$ total concentration of monomer in the reactor, moles/cm^3 emulsion

n chain length of polymer, dimensionless

r radius of particle, cms

R_i rate of initiation, moles/sec.

R_{ia} rate of absorption of radicals in particles, and micelles, moles/sec.cm^2

$[S]_T$ total emulsifier concentration, moles/cm^3 emulsion

S_{wc} critical miscelle concentration, moles/cm^3 water

t time, sec.

$[Tr]_P$ concentration of chain transfer agent inside the particle, moles/cm^3

v volume of particle, cm^3

v_m volume of micelle, cm^3

v_d volume of monomer droplets, cm^3

V_R volume of reactor, cm^3

V_w volume of aqueous phase per cm^3 emulsion, cm^3/cm^3 emulsion

$\bar{\varphi}$ monomer volume fraction in the particles, dimensionless

ψ monomer polymer interaction parameter, dimensionless

γ surface tension, dynes/cms

θ mean residence time, sec.

δ_{ig} Kronecker delta function, 1 for i=J, 0 for i$\neq$J

$\delta(v-v_m)$ Dirac delta function; 1 for v=v$_m$, 0 for v$\neq$v$_m$.

CHAPTER 4

METHOD OF SOLUTION OF THE MODELLING EQUATIONS

In the previous chapter, the modelling equations for emulsion polymerization in a CSTR were presented. These equations contain various material balance equations and the population balance equations. The material balance equations are ordinary differential or nonlinear algebraic equations and are relatively easy to handle. However, the various population balance equations involve different distribution functions like $F(v)$, $f(i,v)$, $L(n,i,v)$ and $G(n,i,v)$ and are complex coupled partial differential-difference equations. Special care is required to decide the line of approach to solve these equations. In this chapter normalisation procedures have been introduced and for convenience the additional nomenclature for this chapter is listed at the end of the chapter.

The aim of solving these equations is to calculate the values of the aforesaid distribution functions. However, generation of these functions numerically for different values of attributes may take exorbitant time. To avoid this, the problem can be approached by making use of the concept of moments of a distribution,

Let $f(x)$ represent a continuous distribution for the attribute x. Then the kth moment of $f(x)$ is defined as

$$\overline{\overline{f}}_k = \int_0^\infty x^k \, f(x) \, dx \qquad (4.1)$$

Similarly if H(i) represents a distribution defined only for discrete values of i, then the kth moment of the discrete distribution is given by

$$\overline{\overline{H}}_k = \sum_{i=i\,min}^{i=i\,max} i^k \, H(i) \qquad (4.2)$$

It can be seen from these definition that, with the knowledge of a distribution function, all of its moments can be calculated. Obviously, a continuous distribution function, say f(x), containing infinite information cannot be exactly specified by a finite set of moments. However, it is possible to represent a distribution function, in terms of finite number of moments to an accuracy sufficient for modelling purposes. One of the methods to achieve this is to represent the distribution function in a series expansion of orthogonal polynomials like Laguerre polynomial [17,30]. The Laguerre polynomial $L_n{}^\alpha(x)$ has the following orthogonal property.

$$\int_0^\infty (x)^\alpha \, \exp(x) \, L_n{}^\alpha(x) \, L_m{}^\alpha(x) \, dx = \frac{\overline{\alpha + n + 1}}{\underline{n}} \, \delta nm \qquad (4.3)$$

where $\delta nm = 1 \qquad$ if n=m $\qquad (4.4)$

$\qquad\qquad\quad = 0 \qquad$ if n≠m

where $L_n{}^\alpha(x) = \sum_{j=0}^{n} (-1)^J \binom{n+\alpha}{n-J} \frac{x^J}{\underline{J}} \qquad (4.5)$

Using the definition and orthogonal property of Laguerre polynomial, as stated above, a normalized distribution function $G(y)$ can be represented in terms of summations involving Laguerre polynomial [40].

$$G(y) = \frac{y^{\alpha} \exp(-y)}{\overline{|\alpha+1}} \sum_{n=0}^{m} Cn^{\alpha} \, Ln^{\alpha}(y) \tag{4.6}$$

The coefficients Cn^{α} are related to the moments $\overline{\overline{G}}_{k}$ of the normalized distribution function $G(y)$ by following expression:

$$C_{n}^{\alpha} = \sum_{j=0}^{n} (-1)^{J} \frac{\lfloor n}{\lfloor n-J \, \lfloor J} \frac{\sqrt{\alpha+1}}{\sqrt{\alpha+J+1}} \overline{\overline{G}}_{J} \tag{4.7}$$

Equation 4.7 follows from the orthogonal properties of L_{n}^{α}. Following Min [30], the parameter α is chosen so that C_{1}^{α} and C_{2}^{α} are zero, which results in rapid convergence. The resulting expression for α is the following:

$$\alpha = -1 + \frac{(\overline{\overline{G}}_{1})^{2}}{\overline{\overline{G}}_{2} - (\overline{\overline{G}}_{1})^{2}} \tag{4.8}$$

The normalized function $G(y)$, used here is related to the absolute distribution function $f(x)$ as follows:

$$f(x)dx = G(y) \, dy \, \overline{\overline{f}}_{o}$$

and

$$y = \frac{\overline{\overline{f}}_{1}}{\overline{\overline{f}}_{o}(\alpha+1)} \tag{4.9}$$

The moments of the two distributions, $f(x)$ and $G(y)$, are related to each other by the following expression

$$\bar{\bar{f}}_k = \bar{\bar{f}}_o \left[\frac{\bar{\bar{f}}_1}{\bar{f}_o (\alpha+1)} \right]^{k-1} \bar{\bar{G}}_k \qquad (4.10)$$

It can be seen from the preceding lines that most of the information contained in a continuous distribution function can be obtained from a finite set of moments of that distribution. The number of moments required to specify a differential distribution depends on the complexity of its shape. It has been found that only 3 or 4 moments are often sufficient to represent a unimodal distribution satisfactorily.

Even in practical cases, it is often impossible to measure complete molecular weight distribution or particle size distribution. Generally it is convenient to measure and use the leading moments of these distributions for quality control. This method involves the use of M_n, M_w or polydispersity index to represent m.w.d. and average size to specify the particle size distribution.

For the reason stated above, the modelling equations given in the previous chapter involving differential distribution functions will be transformed to a set of equations involving finite number of moments of those distributions.

It should be noted that the method to represent a distribution in terms of its moments is useful only for continuous distributions. Obviously a discrete distribution containing a large number of closely spaced points can be approximated by a continuous distribution. Consequently, in

the present study the live polymer chain distribution as
well as the dead polymer chain length distributions which
are actually defined for integer values of n only, are
approximated by a continuous distribution in n and the
aforesaid method is then made use of.

However, a distribution of polymer particles in i
(the number of radicals contained in the particle) is defined
only at a few points. It has been found from experiments
that the number of live radicals contained in a particle
rarely exceeds 10. As a result, the use of moments to
represent the discrete distribution in i is not justifiable.
Hence this distribution will be generated numerically at the
discrete points at which it actually exists (i.e. at i = 0,1,2..).
Since the number of points at which it is defined is small
this technique involves the solution of a relatively small
number of equations.

To start with, the following assumptions will be
made for mathematical simplicity.

1. A distribution in the volume of particles is
generated. The fractional moments in volume (i.e. two third
and one third moment) are used to calculate the total surface
area and the average diameter of the particles. However,
when 'v' is connected with the distribution functions in a
more complex way (like the term arising due to termination
of radicals in particle), the average value of the volume

of particles is used for mathematical simplicity. This approximation is likely to be more exact when the particle size distribution is relatively narrow.

2. The terms representing the chain transfer to transfer agent and initiation in particles will be maintained while developing the sets of equations involving moments of live and dead polymer chain length distribution. However, the values of $[\bar{Tr}]_P$ and $[\bar{I}]_P$ can be evaluated by addition of two material balance equations and thus these mechanisms can be easily accounted for.

3. The value of monomer concentration in particles $(\bar{\varphi})$ can be calculated from equation 3.30. This equation shows that $(\bar{\varphi})$ is in general a function of particle size. However, from experimental studies it has been found that the concentration of monomer in a particle is almost constant and independent of its volume. The values of the concentrations of monomer in a polymer particle are reported in literature for different monomers. Hence, these values can be directly used to solve remaining modelling equations. Most of the results from this study have been obtained by this method. However, the results obtained without making this assumption are also given for comparison.

The modelling equations given in the earlier chapters are not dimensionless. Hence the numerical values of various variables depend upon the system of units to be used. These

equations should firstly be transformed into sets of nondimensional modelling equations. This approach facilitates the use of these equations, since the variables are not confined to any particular system of units. Similarly it facilitates the numerical solution of these equations, since the range of numerical values of various dimensionless variables is relatively narrow. The dimensionless population balance equations should then be transformed into the sets of equations involving moments of the dimensionless distributions.

This procedure has been explained in the following lines, for the case of live polymer chain length distribution.

Consider the population balance equation (3.10) involving live polymer chain length distribution:

$$\frac{\partial L(n,i,v,t)}{\partial t} + \frac{\partial}{\partial v}(r_v \cdot L(n,i,v,t)) = R_{ia}[\frac{a_P}{A_P+A_m}]$$

$$[L(n,i-1,v,t) - L(n,i,v,t) + f(i-1,v,t)\cdot\delta_{n1}]$$

$$+k_d[I]_P \, v\cdot N_A \, [L(n,i-2,v,t) - L(n,i,v,t) + 2\delta_{n1}\cdot f(i-2,v,t)]$$

$$+ \frac{k_o \, a_P}{v}[(i) L(n,i+1,v,t) - (i) L(n,i,v,t)] + k_P[M]_P$$

$$[L(n-1,i,v,t) - L(n,i,v,t)] + k_{ft}[Tr]_P [i \, f(i,v,t) \, \delta_{n1}$$

$$-L(n,i,v,t)] + k_{fm}[M]_P [i \, f(i,v,t) \, \delta_{n1} - L(n,i,v,t)]$$

$$+ \frac{k_{tc} + k_{td}}{N_A v} [\frac{(i+1)(i)}{2} L(n,i+2,v,t) - \frac{(i)(i-1)}{2} L(n,i,v,t)]$$

$$+ \frac{1}{\theta} [L(n,i,v,t)\Big|_{feed} -L(n,i,v,t)] + R_{ia}(\frac{A_m}{A_P+A_m})\delta_{n1} \, \delta_{i1} \, \delta(v-v_m)$$

$$(3.10)$$

At steady state the derivatives with respect to
time can be equated to zero.

Similarly all the particles are assumed to have volume
v_{avg}. Hence the derivatives with respect to $\dot{v}$ also reduce
to zero, and the average volume of the particles v_{avg} can
then be substituted in place of volume, v, in these equations.
Using the definitions of various dimensionless variables and
dimensionless constants given at the end of this chapter
and after further simplifications this equation reduces to

$$0 = C_7[\bar{I}]_W \frac{(\bar{v})^{2/3}}{\bar{F}_{2/3}^{+\bar{m}}} [\bar{L}(n,i-1) - \bar{L}(n,i) + \bar{f}(i-1)\ \delta_{n1}]$$

$$+ C_2[\bar{I}]_P \cdot v[\bar{L}(n,i-2) - \bar{L}(n,i) + 2\delta_{n1}\ \bar{f}(i-2)]$$

$$+ C_3 \cdot (\bar{v})^{1/3}[i\ \bar{L}(n,i+1) - i\ \bar{L}(n,i)]$$

$$+ C_{101}(\bar{\varphi})[\bar{L}(n-1,i) - \bar{L}(n,i)]$$

$$+ C_{63}[\bar{Tr}]_P [\delta_{n1} \cdot (i)\ \bar{f}(i) - \bar{L}(n,i)]$$

$$+ C_{102}(\bar{\varphi}) [\delta_{n1} \cdot i\ \bar{f}(i) - \bar{L}(n,i)] + \frac{C_4}{(\bar{v})}[(i)\ (i+1)$$

$$\bar{L}(n,i+2) - (i)(i-1)\ \bar{L}(n,i)] + \frac{1}{\theta}[\bar{L}(n,i) \bigg\downarrow_{feed} -\bar{L}(n,i)]$$

$$+ C_7[\bar{I}]_W \frac{\bar{m}}{(\bar{F}_{2/3} + \bar{m})} \delta_{i1}\ \delta(v-v_m)\ \delta_{n1}.$$

$$(4.11)$$

This dimensionless population balance equation is then
converted into a set of equations containing moments of
the distribution in n. To do this the equation is multiplied

by n^k and is summed over all possible values of n. After

further simplifications the following equation is obtained:

$$0 = \frac{C_7[\bar{I}]_w (\bar{v})^{2/3}}{(\bar{\bar{F}}_{2/3} + \bar{m})} [\bar{\bar{L}}_k(i-1) - \bar{\bar{L}}_k(i) + \bar{F}(i-1)]$$

$$+ C_2[\bar{I}]_p(\bar{v}) [\bar{\bar{L}}_k(i-2) - \bar{\bar{L}}_k(i) + 2\bar{F}(i-2)]$$

$$+ C_3(\bar{v})^{1/3} [i\,\bar{\bar{L}}_k(i+1) - (i)\,\bar{\bar{L}}_k(i)]$$

$$+ C_{101}(\bar{\phi}) [\sum_{m=1}^{k} \frac{k}{m\ k-m} \bar{\bar{L}}_{k-m}(i)] + C_{63}[\bar{Tr}]_p [i\,\bar{F}(i)$$

$$- \bar{\bar{L}}_k(i)] + C_{102}(\bar{\phi}) [i\,\bar{F}(i) - \bar{\bar{L}}_k(i)] + \frac{C_4}{\bar{v}} [(i+1)(i)\bar{\bar{L}}_k(i+2)$$

$$- (i)(i-1)\bar{\bar{L}}_k(i)] + \frac{1}{\theta} [\bar{\bar{L}}_k(i)_{feed} - \bar{\bar{L}}_k(i)]$$

$$+ C_7[\bar{I}]_w \frac{\bar{m}}{\bar{\bar{F}}_{2/3} + \bar{m}} \delta_{il} \tag{4.12}$$

Other modelling equations can be rendered dimensionless

and the population balance equations can be transformed to a

set of equations involving moments of dimensionless distri-

butions, by using the aforesaid approach. The resultant set

of modelling equations is given on the following page.

Particle Size Distribution:

$$0 = \bar{I}_{avg} \cdot n \frac{\bar{\phi}}{1-\bar{\phi}} \bar{\bar{F}}_{k-1} + C_7 \frac{[\bar{I}]_w (\bar{m})}{(\bar{\bar{F}}_{2/3} + \bar{m})}$$

$$+ \frac{1}{\bar{\bar{\theta}}} [\bar{\bar{F}}_k \Big|_{feed} - \bar{\bar{F}}_k] \qquad k = 0 \text{ to } n \qquad (4.13)$$

Radical Number Distribution:

$$0 = C_7 \frac{[\bar{I}]_w (\bar{v})^{2/3}}{(\bar{\bar{F}}_{2/3} + \bar{m})} [\bar{f}(i-1) - \bar{f}(i)] + C_3 (\bar{v})^{-1/3} [(i+1)\bar{f}(i+1)$$

$$- i \bar{f}(i)] + \frac{C_4}{(\bar{v})} [(i+2)(i+1) \bar{f}(i+2) - (i)(i-1) \bar{f}(i)]$$

$$+ \frac{C_7 [\bar{I}]_w (\bar{m})}{(\bar{\bar{F}}_{2/3} + \bar{m})} \delta_{i1} \delta(v-v_m) + \frac{1}{\bar{\bar{\theta}}} [\bar{f}(i) \Big|_{feed} - \bar{f}(i)]$$

$$+ C_2 (\bar{v})[\bar{I}]_P [\bar{f}(i-2) - \bar{f}(i)] \qquad i = 0 \text{ to } m \qquad (4.14)$$

Live Polymer Chain Length Distribution:

$$0 = C_7 \frac{[\bar{I}]_w (\bar{v})^{2/3}}{(\bar{\bar{F}}_{2/3} + \bar{m})} [\bar{\bar{L}}_k(i-1) - \bar{\bar{L}}_k(i) + \bar{f}(i-1)]$$

$$+ C_2 [\bar{I}]_P (\bar{v}) [\bar{\bar{L}}_k(i-2) - \bar{\bar{L}}_k(i) + 2\bar{f}(i-2)]$$

$$+ C_3 (\bar{v})^{-1/3} [(i) \bar{\bar{L}}_k(i+1) - (i) \bar{\bar{L}}_k(i)] + C_{101} (\bar{\phi})$$

$$[\sum_{m=1}^{k} \frac{k}{m} \frac{k}{k-m} \bar{\bar{L}}_{k-m}(i)] + C_{63}[\bar{T}r]_P [i \bar{f}(i) - \bar{\bar{L}}_k(i)]$$

$$+ C_{102}(\bar{\phi})[(i) \bar{f}(i) - \bar{\bar{L}}_k(i)] + \frac{C_4}{(\bar{v})} [(i+1)(i) \bar{\bar{L}}_k(i+2)$$

$$-(i)(i-1) \bar{\bar{L}}_k(i)] + \frac{1}{\bar{\bar{\theta}}} [\bar{\bar{L}}_k(i) \Big|_{feed} - \bar{\bar{L}}_k(i)]$$

$$+ C_7 [\bar{I}]_w \frac{\bar{m}}{(\bar{\bar{F}}_{2/3} + \bar{m})} \delta_{i1} \qquad k = 0 \text{ to } n; \ i = 0 \text{ to } m \qquad (4.12)$$

Dead Polymer Chain Length Distribution:

$$0 = \frac{C_7 [\bar{I}]_w (\bar{v})^{2/3}}{(\bar{F}_{2/3} + \bar{m})} [\bar{\bar{G}}_k(i-1) - \bar{\bar{G}}_k(i)]$$

$$+ \frac{C_3}{(\bar{v})^{1/3}} [(i+1) \bar{\bar{G}}_k(i+1) - (i)\bar{\bar{G}}_k(i)] + C_2[\bar{\bar{I}}]_P(\bar{v})$$

$$[\bar{\bar{G}}_k(i-2) - \bar{\bar{G}}_k(i)] + \frac{C_4}{(\bar{v})} [(i+2)(i+1) \bar{\bar{G}}_k(i+2)$$

$$-(i)(i-1) \bar{\bar{G}}_k(i)] + \frac{C_{403}}{(\bar{v})} [\bar{f}(i+2) \sum_{p=0}^{k-1} \frac{\lfloor k}{\lfloor k-p \ \lfloor p} g_p(i+2) g_{k-p}(i+2)]$$

$$+ C_{102}(\bar{\varphi})[\bar{\bar{L}}_k(i)] + \frac{C_{404}}{(\bar{v})} [(i+1) \bar{\bar{L}}_k(i+2)] + C_{63}[\bar{Tr}]_P \bar{\bar{L}}_k(i)]$$

$$+ \frac{1}{\bar{\bar{\Theta}}} [\bar{\bar{G}}_k(i) \Big|_{feed} - \bar{\bar{G}}_k(i)]$$

$$\begin{aligned} k &= 0 \text{ to } n \\ i &= 0 \text{ to } m \end{aligned} \qquad (4.15)$$

Emulsifier Balance:

$$0 = \frac{1}{\bar{\bar{\Theta}}} [[\bar{\bar{S}}_T] \Big|_{feed} - [\bar{S}]_T] \qquad (4.16)$$

Initiator Balance:

$$0 = - [\bar{I}]_w + \frac{C_{56}}{\bar{\Theta}} [[\bar{\bar{I}}]_w \Big|_{feed} - [I]_w] \qquad (4.17)$$

Monomer Balance:

$$0 = \frac{1}{\bar{\bar{\Theta}}} [[\bar{M}]_T \Big|_{feed} - [\bar{M}]_T] + C_{53}(\bar{\varphi}) \bar{\bar{F}}_0 \cdot \bar{i}_{avg} \qquad (4.18)$$

Micelle Balance:

$$0 = -\bar{m} + C_{50} [\bar{S}]_T - C_{51} - \bar{\bar{F}}_{2/3} - C_{52} \bar{m}_d \qquad (4.19)$$

Monomer Droplets Balance:

$$0 = -\bar{\bar{m}}_d + C_{54}[\bar{M}]_T - C_{55} \cdot \bar{\bar{F}}_1 \cdot (\bar{\varphi}) - C_{556}(\bar{\bar{m}}) \tag{4.20}$$

Expression for $(\bar{\varphi})$:

$$0 = (1-\bar{\varphi}) + \ln(\bar{\varphi}) + \mathbf{\Psi}(1-\bar{\varphi})^2 + C_{65} \frac{\bar{\bar{F}}_0}{\bar{\bar{F}}_{1/3}} \tag{4.21}$$

It should be noted that the expressions 4.12, 4.14, 4.15 are functions of k and i. Similarly the expression 4.13 is a function of k. Thus, these expressions indeed represent sets of equations depending on different values of k and i.

The particle size distribution is continuous and the distribution in n can be assumed to be similar to a continuous distribution. Hence the different moments are used in the aforesaid sets of equations to represent them. However, it can be seen from these expressions that different fractional moments with respect to volume (i.e. $\bar{\bar{F}}_{1/3}$ and $\bar{\bar{F}}_{2/3}$) need to be evaluated separately. This can be done by expressing the differential distribution in volume in terms of first 5 integer moments as described previously in equations 4.4 to 4.10. Any fractional moment $\bar{\bar{F}}_t$ can then be calculated from the differential distribution. The resulting equation expressing the fractional moment in terms of first 5 integer moments is given by

$$\bar{\bar{G}}_t^x = \frac{\overline{|\alpha + t + 1}}{|\overline{\alpha+1}} + \frac{1}{|\overline{\alpha+1}} \sum_{n=3}^{m} c_n^\alpha \sum_{k=0}^{n} (-1)^k$$

$$x \frac{\overline{|n + \alpha + 1} \; \overline{|\alpha+t+k+1}}{\overline{|n-k+1} \; \overline{|\alpha+k+1} \; \overline{|k+1}} \tag{4.22}$$

The distribution in i, however, cannot be approximated to behave like a continuous distribution, as explained previously. Hence, the equations inolving i (i.e. 4.12, 4.14 and 4.15) are solved for different discrete values of i. These equations, however, involve advanced terms calculated at (i+1) and (i+2). Thus the set of n equations will involve n+2 variables. However, as stated earlier, the number of radicals in a particle is generally limited to a small value. In other words, after a sufficiently high value of i the terms evaluated at (i+1) and (i+2) tend to become zero. Thus the advanced terms requiring evaluation of (i+1) and (i+2) can be approximated to zero and the set of n equations can be solved for remaining n variables.

Method of Solution:

The modelling equations given earlier can be divided into 3 groups. The first group of equations contains various material balance equations, the equation for $(\bar{\varphi})$, the equations involving moments of particle size distribution and the equations involving radical number distribution. Equations 4.13, 4.14, 4.18, 4.19, 4.20 and 4.21 are interdependent and should be solved simultaneously. Equations 4.16 and 4.17 can be solved separately. The solution of this first group of equations gives the values of the various dimensionless concentration terms, radical number distribution and moments of particle size distribution. Using these values the second

group of equations involving moments of live polymer chain length distribution can be solved to obtain $\overline{\overline{f}}_k(i)$ for different values of i and k. These answers can further be utilized to solve the third group of equations involving dead polymer chain length distribution to obtain $\overline{\overline{G}}_k(i)$ for different values of i and k.

The basic aim of solving these modelling equations was to evaluate the properties of the output stream from the reactor. These properties include conversion, rate of polymerization in the reactor, number of particles per cm^3 of emulsion, particle size distribution and the molecular weight of the resultant polymer. The conversion, rate and number of particles per cm^3 emulsion can be calculated by knowing the values of $[\overline{M}]_T$ and $[\overline{\overline{F}}_5]$.

The expressions to obtain these are given below:

$$X = \frac{[\overline{M}_T] \Big|_{feed} - [\overline{M}]_T}{[\overline{M}]_T \Big|_{feed}} \tag{4.23}$$

$$Rate = \frac{[\overline{M}]_T \Big|_{feed} - [\overline{M}]_T}{\theta} \times \frac{[I_o]_w\, k_p\, d_m}{N_A \cdot d_P \cdot v_m} \tag{4.24}$$

$$N_o = [\overline{\overline{F}}_o] \cdot m_o \cdot N_A \tag{4.25}$$

The particle size distribution of the latex can be generated from its moments $\overline{\overline{F}}_k$ by using equation 4.22. The average volume and average diameter of these latex particles can also be evaluated from these moments.

$$V_{avg} = \frac{\bar{\bar{F}}_1}{\bar{\bar{F}}_0} \tag{4.26}$$

$$D_{number} = \frac{\bar{\bar{F}}_{1/3}}{\bar{\bar{F}}_0} \times \left(\frac{6}{\pi}\right)^{1/3} \tag{4.27}$$

$$D_{vol} = \left(V_{avg} \times \frac{6}{\pi}\right)^{1/3} \tag{4.28}$$

The number average as well as weight average molecular weight and the polydispersity index can be calculated from moments of dead polymer chain length distribution. These expressions are given below:

$$M_n = \frac{\bar{\bar{GT}}_1}{\bar{\bar{GT}}_0} \tag{4.29}$$

$$M_w = \frac{\bar{\bar{GT}}_2}{\bar{\bar{GT}}_1} \tag{4.30}$$

where

$$\bar{\bar{GT}}_s = \sum_{i=0}^{m} \bar{\bar{G}}_s(i) \tag{4.31}$$

$$PDI = \frac{M_w}{M_n} \tag{4.32}$$

The entire molecular weight distribution of the resultant polymer can also be generated from these moments by using the procedure outlined earlier in equations 4.4 to 4.10 and 4.22.

Description of Computer Program:

The equations in group 1 are subdivided in two sub-groups, first containing equations 4.16 to 4.21 and 4.13 while the second comprises of equation 4.14. For the solution

of the first subgroup the feed conditions and initial guesses
for variables as well as initial guess for $\bar{i}_{avg}$ is needed.
This group of equations can then be solved to give values
of different dimensionless concentrations and the moments
of particle size distribution. These values are then fed
to the second subgroup which now becomes a set of linear
equations. The starting value of m=5 is used. The values
of $\bar{f}(m+1)$ and $\bar{f}(m+2)$ are assumed to be zero and the equations
are solved to get values of $\bar{f}(i)$; i=1,m. An increment of
two is then made in m and (m+2) equations are again solved,
assuming $\bar{f}(m+3)$ and $\bar{f}(m+4)$ to be equal to zero. The values
of $\bar{f}(i)$, i=1, m+2 are then obtained. The values of $\bar{f}(m+1)$
and $\bar{f}(m+2)$ are checked to be less than a small constant z
$(10^{-7}$ was used). If these values are not less than z an increment
of two is again made in m and the set of equations is again
solved. The process is repeated until the last two variable
values of a set are less than z, thus checking the assumption
made in the solution of previous set. The value of $\bar{i}_{avg}$ is
then calculated. This improved value of $\bar{i}_{avg}$ is then
compared with the previous value of $\bar{i}_{avg}$ and is used together
with the values of different variables as the improved guesses
to solve the equations in the first subgroup. The process
is repeated till the difference between two values of $\bar{i}_{avg}$
is less than a particular convergence criterion.

The equations in the first subgroup are highly nonlinear
in nature. The method suggested by Brown has been employed

to solve these equations. The details of this method were
taken from reference [31]. It has been found that it is
not possible to give good initial guesses for these equations,
without which the method does not tend to converge to the
solution. Because of this reason an objective function
consisting of sums of absolute values of RHS of the aforesaid
equations is formed. This objective function is then minimized
with respect to the aforesaid variables using the multivariable
constrained optimization technique of Box. The details of
this technique are given in references [32] and [33]. Relatively
worse initial guesses can be used for this optimization
procedure. The values of different variables from the opti-
mization procedure are then used as the initial guesses for
solving the equations of first group. The solution of linear
equations is done by standard matrix inversion procedure.
The subprogrammes needed for solution of this group are NONLIN,
BACK, AMAX1, AMIN1, FUNX, CONSX, CHECH, CENTR, FUNC, CONST,
MATIN, SET1, LINEAR, GOLDEN, CONFR and OBJECT.

The answer from first group of equations are then used
for the solution of equations in second group. While solving
the radical number distribution (i.e. second subgroup of group
1), the assumption made for solving a set of equations was
immediately checked by solving an increased number of equations.
Hence when the convergence was ultimately achieved, the
assumptions made in the solution of last but one set of

equations (containing say, 'p' equations) was verified by the solution of last set of equations. Hence these 'p' equations should, in principle, be solved for live and dead polymer chain length distribution.' However, the solution of live polymer chain length distribution for f(o) type of particles is meaningless. Hence 'p-1' equations are solved for live polymer chain length distribution. However, for dead polymer chain length distribution 'p' equations need to be solved. The equations are linear and are solved by the same method employed earlier. The subprogrammes required for this part are SET2, MATIN and CONFR.

The answer from 2nd group of equations is then used to solve the equations in 3rd group. The method of solution of these equations is similar to that employed for the 2nd group. The answers from this part give the values of $\overline{\overline{G}}_k(i)$ for different values of i and k. The subprogrammes required for this portion are SET3, MATIN and CONFR.

After the solution of modelling equations various properties of output stream are calculated by using equations 4.23-4.32. Cumulative particle size distribution, live polymer chain length distribution, molecular weight distribution are then generated. The subprogrammes required for this portion include DISTR, DIFF2, CONFR and QG10.

The diagram showing the scheme to solve the modelling equations is given on the next page.
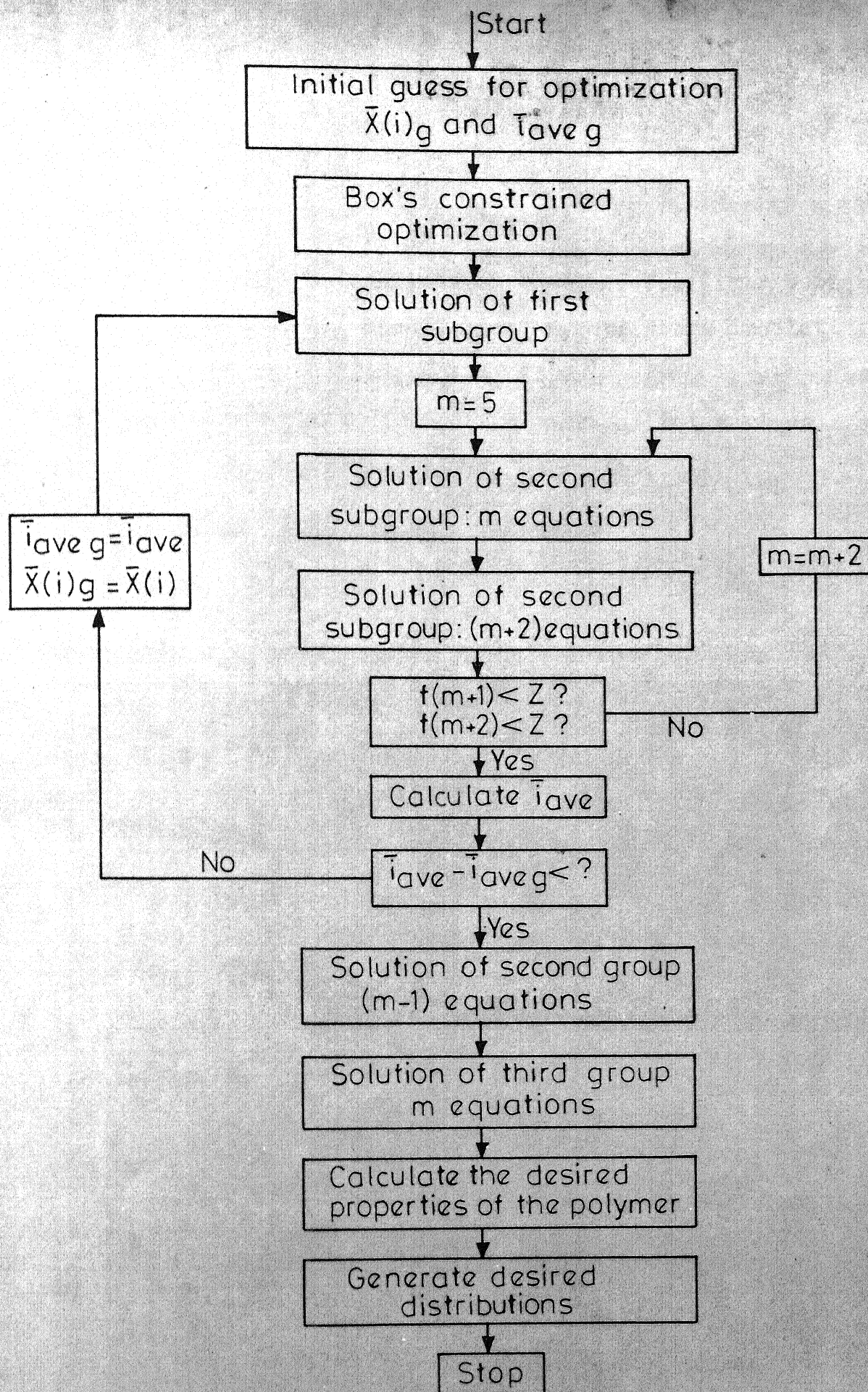
Fig. 1 — Scheme for solution of modelling equations.

NOMENCLATURE

(1) m_0 = nondimensionalization constant for distribution functions, micelle concentration and droplet concentration, moles/lit.

(2) $[\bar{I}_0]_W$ = nondimensionalization constant for various concentration terms, moles/lit.

Definitions of Dimensionless Variables:

1. $[\bar{S}]_T = \dfrac{[\bar{S}]_T}{[\bar{I}_0]_W}$

2. $[\bar{I}]_W = \dfrac{[I]_W}{[\bar{I}_0]_W}$

3. $[\bar{M}]_T = \dfrac{[M]_T}{[\bar{I}_0]_W}$

4. $[\bar{m}] = \dfrac{m}{m_0}$

5. $(\bar{m}_d) = \dfrac{m_d}{m_0}$

6. $\bar{f}(i) = \dfrac{f(i)}{m_0}$

7. $\bar{L}(n,i) = \dfrac{L(n,i,v_{avg})}{m_0}$

8. $\bar{\bar{L}}_k(i) = \displaystyle\sum_{n=1}^{\infty} n^k \, \bar{L}(n,i)$

9. $\bar{\bar{F}}_k = \qquad \bar{v}^k \, \bar{F}(\bar{v}) \, d\bar{v}$

10. $\bar{F}(\bar{v}) = \dfrac{F(v)}{m_0}$

11. $\bar{G}(n,i) = \dfrac{G(n,i,v_{avg})}{m_o}$

12. $\bar{\bar{G}}_k(i) = \displaystyle\sum_{n=1}^{\infty} n^k\, \bar{G}(n,i)$

Defintions of Dimensionless Constants:

1. $C_7 = 2 \cdot k_d \cdot N_A \cdot d_P (I_o)_w \cdot V_m \cdot V_w \cdot f / (K_P \times d_m \times m_o)$

2. $C_2 = \dfrac{k_{iP}}{k_P}\, N_A^2\, v_m^2\, [I_o]_w \cdot \dfrac{d_P}{d_m} \cdot f$

3. $C_3 = \dfrac{k_o}{k_P} \cdot N_A \cdot \dfrac{d_P}{d_m} \cdot (v_m)^{2/3}\, (36\pi)^{1/3}$

4. $C_4 = \dfrac{k_{tc} + k_{td}}{2 k_P} \cdot \dfrac{d_P}{d_m}$

5. $C_{50} = \dfrac{[I_o]_w}{(36\pi)^{1/3}} \quad \dfrac{CE}{(36\pi)^{1/3} \cdot v_m^{2/3} m_o \cdot N_A}$

6. $C_{51} = \dfrac{S_{wC}\, CE}{(36\pi)^{1/3} \cdot v_m^{2/3} m_o} \cdot \dfrac{1}{N_A}$

7. $C_{52} = \dfrac{v_d^{2/3}}{v_m^{2/3}}$

8. $C_{53} = \dfrac{N_A \cdot v_m \cdot d_P \cdot m_o}{[I_o]_w\, M_w}$

9. $C_{54} = \dfrac{[I_o]_w\, V_{mol}}{N_A \cdot v_d \cdot m_o}$

10. $C_{556} = \dfrac{[M]_m \cdot v_m \cdot V_{mol}}{v_d}$

11. $C_{55} = \dfrac{v_m}{v_d}$

12. $C_{56} = \dfrac{k_P \cdot d_m}{k_d \cdot N_A \cdot d_P \cdot v_m}$

13. $\quad c_{63} = \dfrac{k_{ft}}{k_P} \cdot [I_o]_w \cdot N_A \cdot \dfrac{d_P}{d_m} \cdot v_m$

14. $\quad c_{65} = \left(\dfrac{4\pi}{3v_m}\right)^{1/3} \dfrac{2\, V_{mol} \cdot \gamma}{RT}$

15. $\quad c_{101} = \dfrac{d_P \cdot v_M \cdot N_A}{d_m \cdot V_{mol}}$

16. $\quad c_{102} = \dfrac{k_{fm}\, N_A\, d_P\, v_m}{k_P\, d_m\, V_{mol}}$

17. $\quad c_{403} = \dfrac{k_{tc}}{2k_P} \cdot \dfrac{d_P}{d_m}$

18. $\quad c_{404} = \dfrac{k_{td}}{2k_P} \cdot \dfrac{d_P}{d_m}$

CHAPTER 5

RESULTS AND DISCUSSION

The modelling equations developed in the earlier
chapters have been solved by using the approach outlined
in the previous chapter. The model has been applied for
the reaction conditions reported by DeGraff and Poehlein [21].
Their work is one of the most readily available and
comprehensive work on emulsion polymerization in a CSTR.
The experimental set-up used by these authors consisted of
a jacketed stirred tank reactor fitted with a temperature
controller. Nitrogen atmosphere was maintained inside the
reactor. The reactants were fed in two different streams;
first stream comprising of monomer (styrene) and the other
comprising of aqueous solution of initiator (Ammonium per-
sulfate) and surfactant (sodium lauryl sulfate). The effluent
stream was collected in a stirred vessel containing hydro-
quinone inhibitor and was then used for experimental measure-
ments. They have also developed a theoretical model, using
residence time distribution (RTD) approach, to obtain particle
size distribution, rate of polymerization, number of particles
and number average molecular weight of the polymer product.
They have carried out experiments at various reaction
conditions and have tried to compare the results with the
values obtained from their model. Computations using the
model developed here will be compared with their results.

Various constants used for the solution of the modelling equations are listed on the next page.

As has been stated earlier the chain transfer to transfer agent and the initiation in particles have been neglected in the present work. These mechanisms can, in general, be neglected if the monomer feed is free from transfer agent or trace impurities and if it is insoluble in water. Hence the values of k_{ft} and k_{iP} are assumed to be zero. k_{td} is also assumed to be zero, since for styrene the contribution due to termination by disproportionation is usually negligible compared to that with termination by combination [37]. It has found from earlier studies on batch systems that desorption from particles play a relatively small part in emulsion polymerization of styrene [15]. Hence the value of k_o is assumed to be zero. It has been reported that the micelles swell and approximately double their size when saturated with monomer [36]. This implies that approximately 75 per cent of the volume of a saturated micelle is filled with monomer which results in the value of $[M]_m$ as 0.0063.

The reaction was carried out by DeGraff and Poehlein [21] at several temperatures and surfactant concentrations. Only the runs at $70^{\circ}C$ and with a surfactant concentration of 2.79 gms/ 100 gms water were selected for simulation. The monomer to water ratio in the reactor feed was 0.54. From this ratio, V_W is calculated to be 0.6494. The value of $(\bar{\varphi})$ was

	Constant	Units	Value	Ref.No.
1.	k_P	cm^3/mole.sec	2.650×10^5	[21]
2.	k_o	cms/sec	0.0	assumed
3.	k_d	1/sec	2.33×10^{-5}	[21]
4.	k_{iP}	1/sec	0.0	assumed
5.	k_{td}	cm^3/mole sec	0.0	assumed
6.	k_{tc}	cm^3/mole sec	7.2×10^9	[21]
7.	k_{ft}	cm^3/mole sec	0.0	assumed
8.	k_{fm}	cm^3/mole sec	21.2	[34]
9.	d_P	gms/cm^3	1.0526	[21]
10.	d_m	gms/cm^3	0.8780	[21]
11.	v_m	cm^3	65.4498×10^{-21}	[13]
12.	V_{mol}	cm^3/gm.mole	118.45	[21]
13.	v_d	cm^3	4.1888×10^{-12}	assumed
14.	M_w	gm/mole	10^4	-
15.	$[M]_m$	moles/cm^3	0.0063	[36]
16.	CE	cm^2/mole	$3.672 \times 10^{+9}$	[21]
17.	γ	dynes/cms	7.50	[10]
18.	ψ	dimensionless	0.256	[10]
19.	S_{wc}	moles/cm^3 water	8.9×10^{-6}	[35]

reported by DeGraff and Poehlein [21] to be 0.615 and has been used for most of the simulations.

Calculations were made using the aforesaid constants at various reaction conditions and the results were compared with the theoretical predictions and experimental results obtained by DeGraff and Poehlein [21].

Particle Size Distribution:

The cumulative particle size distributions calculated at different conditions have been presented in Figures 2 and 3. It can be seen from these plots that the results are practically identical to the theoretical results obtained by DeGraff and Poehlein.

As can be seen from the figures, the breadth of the particle size distribution increases with increasing residence time. With the increase in the mean residence time, the distribution of times spent in the reactor broadens. Further the rate of volume growth (r_v) will also increase due to the increase in the value of i_{avg}. Both these effects cause a broader particle size distribution.

In the RTD approach it is assumed that a fraction of the micelles fed to the reactor do not get activated and the rest of the micelles are converted into particles at the same instant. Hence all the particles should have the same birth time. The RTD approach followed by DeGraff and Poehlein then requires the knowledge of only two mechanisms; (1) kinetics of volume growth of particles and (2) RTD of the reactor.
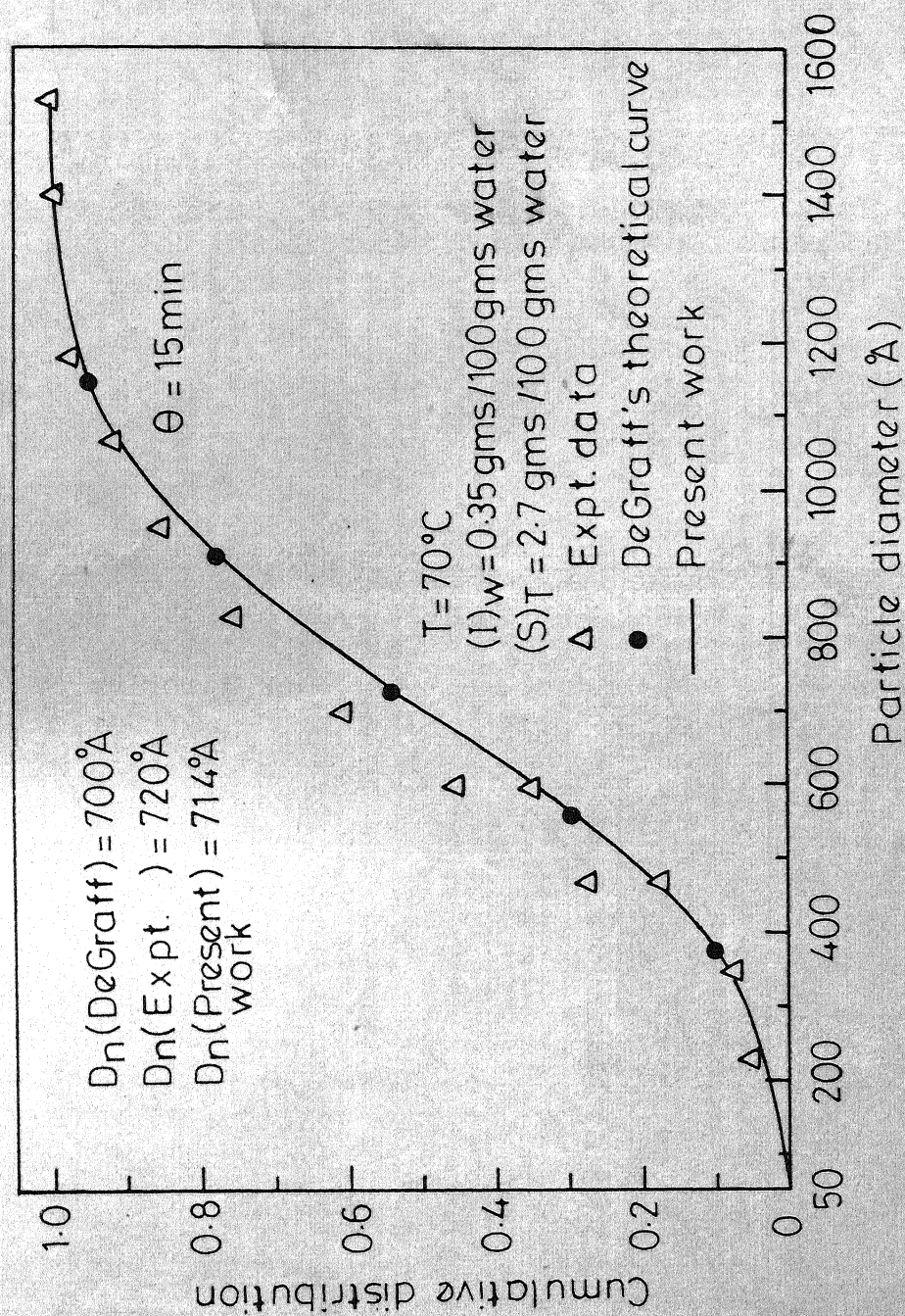
Fig. 2 - Particle size distribution.
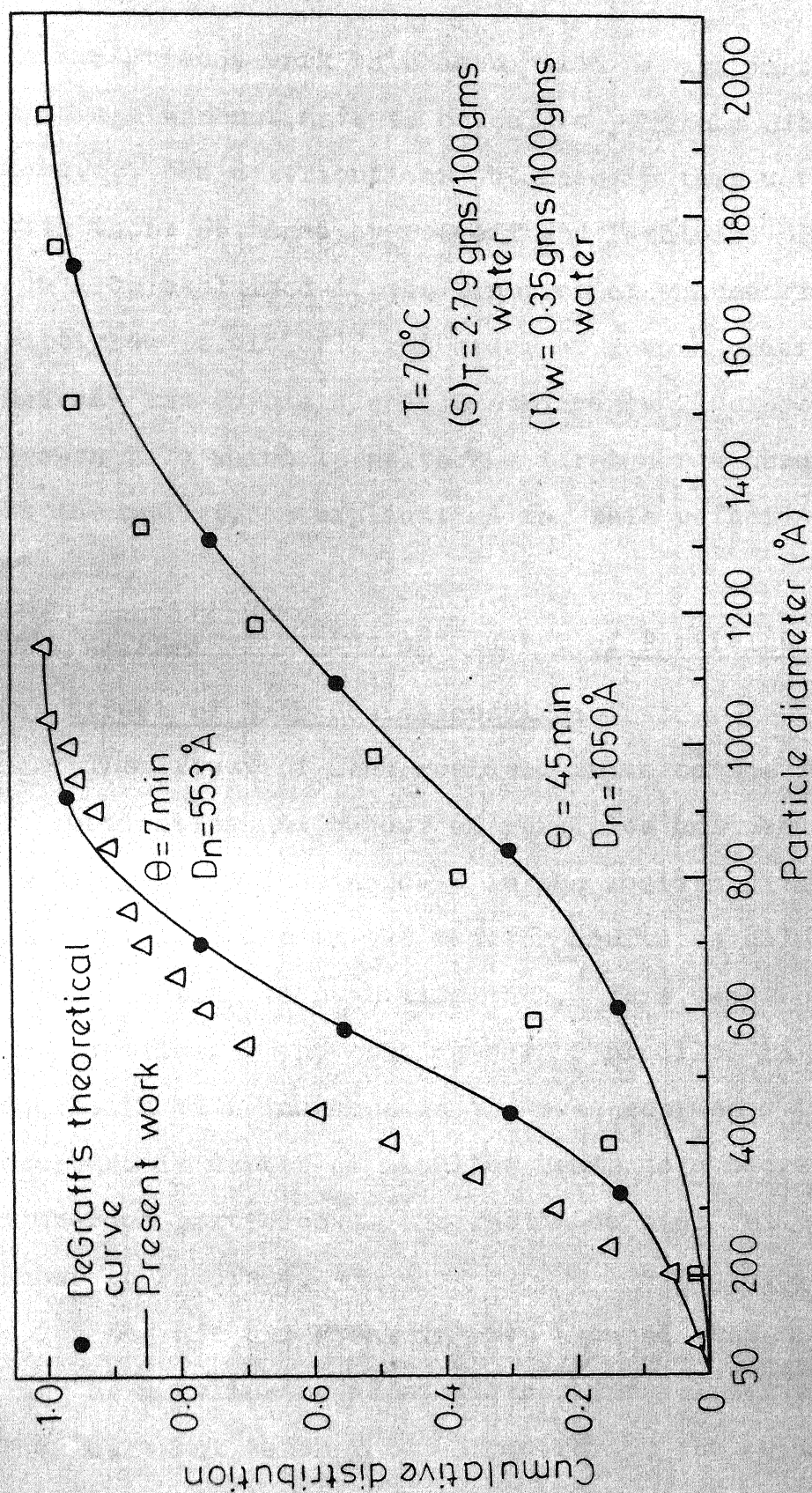
Fig. 3 — Effect of mean residence time on particle size distribution.

In the present work this assumption of all particles being of the
same age was not made to calculate particle size distribution.
However, the distributions obtained in this work coincide
with those obtained by DeGraff and Poehlein. Such coincidence
could be explained if the kinetics of volume growth of
particles is either first order or zeroth order. However,
DeGraff and Poehlein used Stockmayer's[2] expression for
growth rate which is neither a first nor a zeroth order process.
At the moment, an explanation for this coincidence cannot be
offered.

Rate of Polymerization and Number of Particles:

(a) Effect of Mean Residence Time:

The effect of mean residence time on the rate of
polymerization and number of particles has been presented
in Figure 4. With increase in the residence time, the average
size of particles in the reactor increases and requires more
amount of soap to stabilize them. This soap is supplied by
the micelles. Hence the number of micelles in the reactor
decreases with increase in the mean residence time. The
decrease in number of micelles leads to a decrease in the
number of particles as the residence time is increased as
shown in Figure 4.

DeGraff and Poehlein have assumed that soap is present
only as micelles or as stabilizer for polymer particles.
They have not taken into consideration the emulsifier present
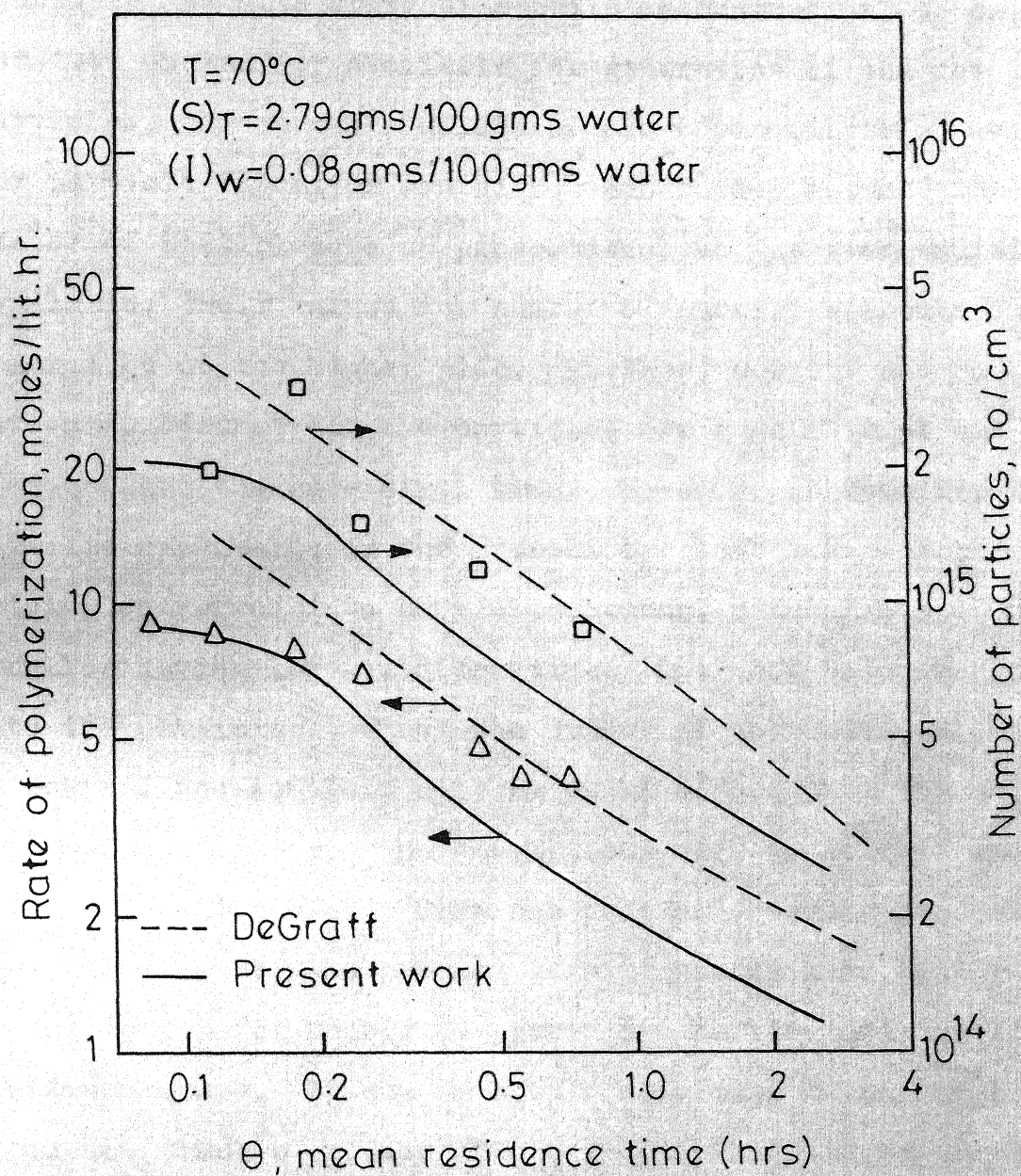
Fig. 4 – Effect of mean residence time on number of particles and rate of polymerization.

on the surface of monomer droplets and that present in
aqueous phase upto critical micelle concentration, to calculate
the free emulsifier available for generation of the new
particles. Hence their values of the free emulsifier available
for particle formation are high. Since they assumed the
number of particles to be proportional to the free emulsifier
available, their values for number of particles should in
general be on the higher side. Further, DeGraff and Poehlein
have assumed that all the particles are generated at the same
time andhave the same birth time. However, in reality, the
micelles may reside in the reactor for some time before
getting converted into particles. Thus, actually, some of the
micelles having low residence times, will not be converted
into the particles. Hence the number of particles calculated
by DeGraff and Poehlein will be still higher than the actual
number of particles. The assumptions made by DeGraff and
Poehlein, stated above, have not been made in the present
work. Therefore, the present results should give better
approximation for number of particles, particularly at low
residence times. It can be seen from Figure 4 that at low
residence times a considerably better fit to the experimental
data has been obtained using the present theory, while the
results obtained by DeGraff and Poehlein are far from the
experimental points. The deviation from theoretical predictions
at high residence times is however inexplicable.

The rate of polymerization will be directly proportional to the number of particles times $\bar{i}_{avg}$ as can be seen from 3.28 and 4.24. Hence an explanation similar to that given in the previous paragraph can be given about effect of mean residence time on the rate of polymerization also. The number of particles decreases while the average number of radicals per particle increases with increase in mean residence time. The variation of rate which involves the product of these two terms in given in Figure 4. The almost identical variation of the rate of polymerization and number of particles implies that the average number of radicals in a particle does not vary much with increase in the mean residence time.

(b) Effect of Initiator Concentration:

Figure 5 shows the effect of variation of initiator concentration on M_n and M_w. When the usual values of mean residence time and concentrations of initiator and soap are used the rate of initiation of radicals is considerably high compared to the free emulsifier available. Hence under these reaction conditions the number of particles in the reactor is considerably larger than the number of micelles. When higher concentrations of initiator are used higher rates of generation of radicals are obtained. However, most of these radicals will be captured by the particles since the total area of the particles is considerably larger than the total area of
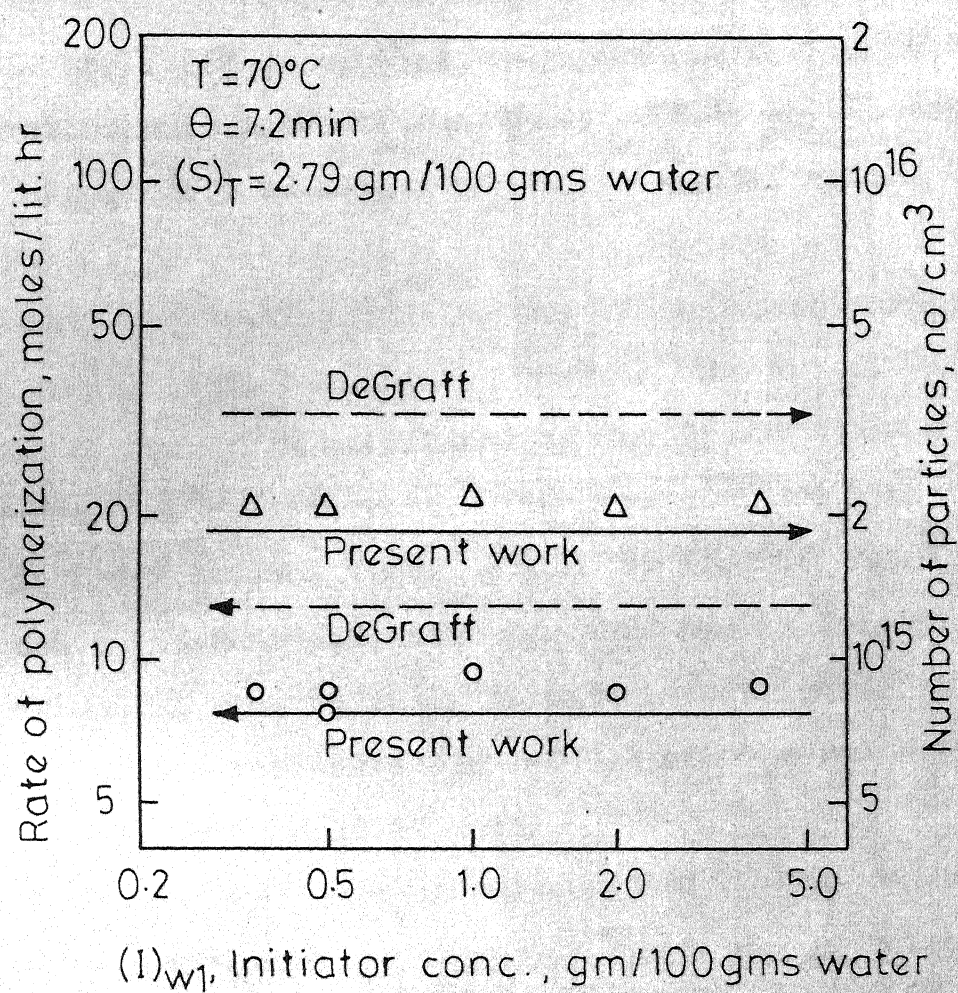
Fig. 5 —Effect of initiator conc. on rate of polymerization and number of particles.

the previous paragraph the value of number of particles
is almost constant with respect to initiator concentration.
Hence the rate of polymerization, which is proportional to
the product of number of particles and the average number of
radicals per particle, should not vary significantly with
changes in initiator concentration. This behaviour can be
seen from Figure 5. The present theory follows the experi-
mental data reasonably well.

Molecular Weights:

(1) Effect of Mean Residence Time:

The effect of mean residence time on M_n and M_w is shown
in Figure 6. With increase in mean residence time there is an
increase in the particle size and hence the total rate of
entry of radicals into particles increases. This decreases
the time of growth of radicals inside the particles. Moreover,
the average number of radicals present in a particle increases
with size leading to the faster termination and lower times
of chain growth. Due to these reasons the live chains
present inside the particles grow for shorter times resulting
in lower molecular weights. This effect can be seen from
Figure 6.

DeGraff and Poehlein [21] did not consider the effect
due to chain transfer to monomer for calculating M_n. However,
this phenomena in general will always be operative and will
tend to bring down the molecular weights of the polymer. This
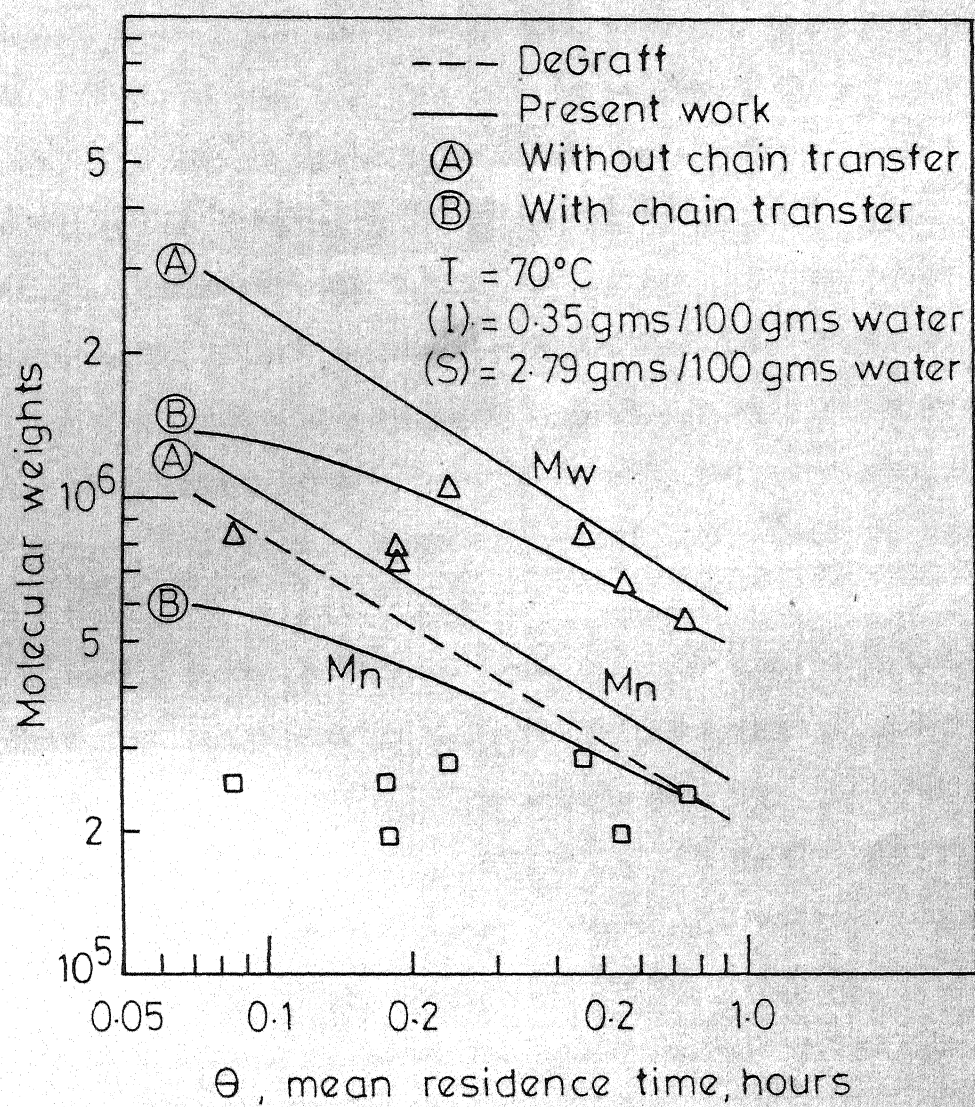
Fig. 6 - Effect of mean residence time on molecular weights.

can be seen from Figure 6 where the variation of molecular
weights have been shown with and without considering chain
transfer to monomer. The values obtained by the present
theory are found to be better approximations to the experi-
mental results than those obtained by DeGraff and Poehlein's
theory. However, the values still do not agree with the
experimental results satisfactorily at low residence times.
Better fit could be obtained by taking into consideration
the chain transfer to impurities and the termination by
disproportionation. These processes will tend to bring down
the molecular weight further.

DeGraff and Poehlein from the RTD approach could not
obtain the weight average molecular weight of the product.
However, the present theory allows the calculation of M_w from
moments of dead polymer chain length distribution. These
values are plotted in Figure 6 with and without considering
chain transfer to monomer. It can be seen that these values
compare fairly well with the experimental results.

(2) Effect of Initiator Concentration:

The effect of initiator concentration on molecular
weights has been presented in Figure 7. As the initiator
concentration increases the rate of radical absorption in
the particles will increase. With the increase in the
frequency of radical absorption in a particle, the live
chains inside the particles are terminated faster, resulting
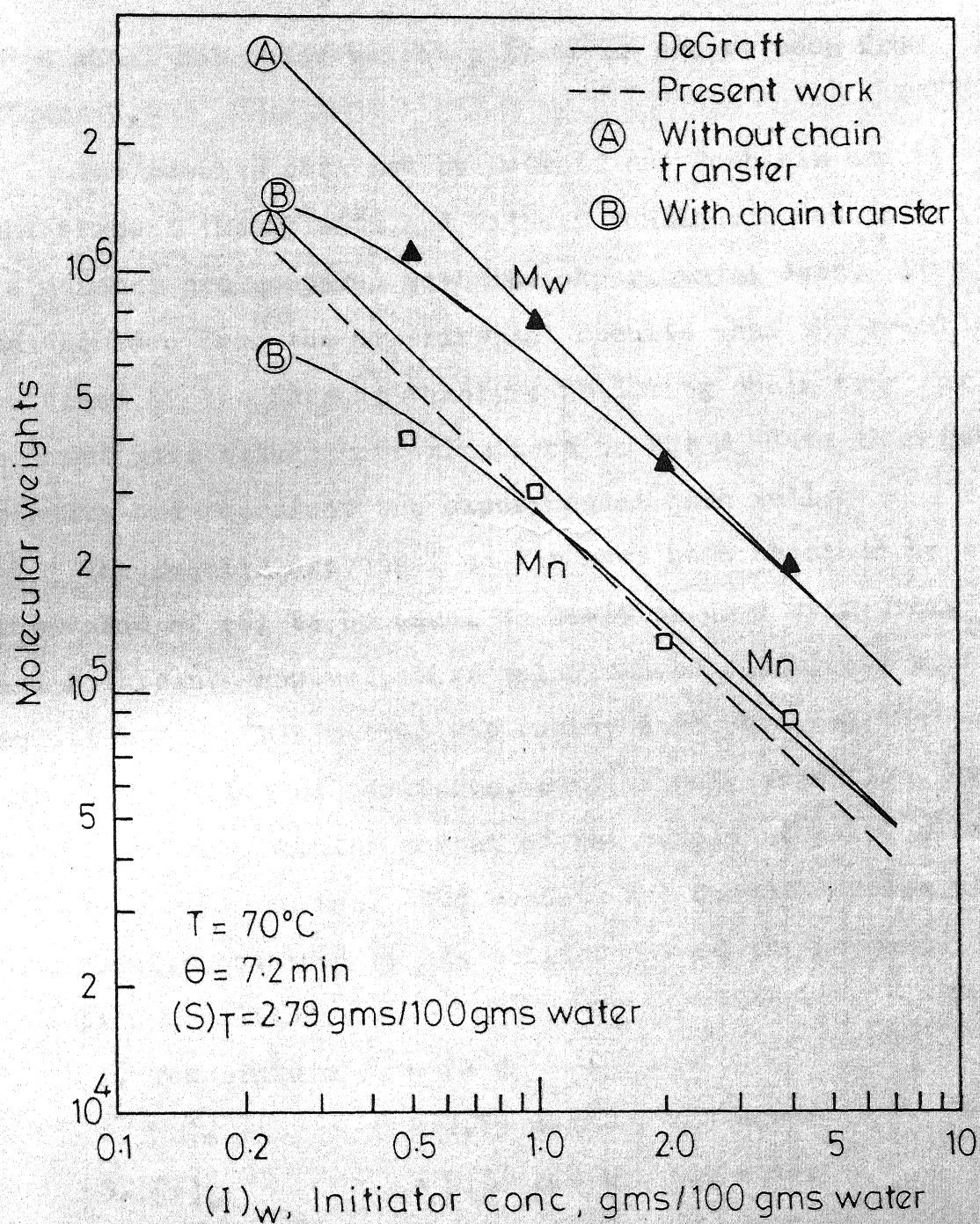
Fig. 7 - Effect of initiator concentration on molecular weights.

in a small molecular weight polymer as can be seen from
Figure 7.

The results obtained by DeGraff and Poehlein and by
the present theory with and without considering chain transfer
to monomer are compared with the experimental data. It
can be seen from the experimental results that the results
obtained by the present theory considering chain transfer to
monomer give better approximations to the M_n data than DeGraff's
results and represent the experimental data well.

The results discussed so far have been obtained by using
the value of ($\bar{\phi}$) to be equal to 0.615 as used by DeGraff
and Poehlein. However, this value can be calculated by using
equation 3.31. This method was employed to obtained the
values of number of particles, rate of polymerization, average
size and the molecular weight of the resultant polymer latex
in simulating one run. The results are compared below with
the results obtained by the earlier method at the same
reaction conditions:

1. Temperature $= 70^{\circ}C$

2. $[S]_T$ $= 2.79$ gms/100 gms water

3. $[I]_W$ $= 0.35$ gms/100 gms water

4. θ $= 15$ minutes

		First Approach	Second Approach
1.	$\bar{\varphi}$	0.615	0.801 (calculated from 4.21)
2.	Average diameter (number basis Å)	729	986
3.	Average diameter (volume basis Å)	817	1104
4.	Number of particles per cm^3 of emulsion	1.09×10^{15}	6.12×10^{14}
5.	Rate of polymerization moles/hr	4.7	3.47
6.	M_n	5.46×10^5	4.006×10^5
7.	M_w	1.32×10^6	9.392×10^5
8.	Polydispersity index	2.43	2.344

It can be seen from the table given above that the value of $\bar{\varphi}$ obtained by the latex method (i.e. $\bar{\varphi}$ = 0.801) is different from the one used in earlier method (i.e. $\bar{\varphi}$ = 0.615). Because of the higher value of $\bar{\varphi}$ obtained in the latter method the rate of volume growth would be higher and hence the average particle size obtained is higher. Because of the higher size of particles, they require more soap for stabilization and hence less amount of soap is present as micelles. This results in lower value for number of particles by the latter method. Similarly, because of the larger size of the particles the frequency of radical entry in a particle is increased, leading to shorter growth times for the radicals

present in the particle. This results in the lower values of molecular weights as reported in the table.

The aforesaid comparison is made without considering chain transfer to monomer. The comparison was also made by taking chain transfer into consideration. The results are similar to the case where chain transfer is not considered. The molecular weights obtained by both the approaches are, however, smaller than the first case.

		First Approach	Second Approach
1.	$\bar{\varphi}$	0.615	0.819 (Calculated from 4.21)
2.	Average diameter (number basis $\overset{\circ}{A}$)	714	986
3.	Average diameter (volume basis $\overset{\circ}{A}$)	799	1104
4.	Number of particles per cm^3 emulsion	1.14×10^{15}	6.12×10^{14}
5.	Rate of polymerization moles/hr	4.78	3.47
6.	M_n	3.925×10^5	3.092×10^5
7.	M_w	9.215×10^5	7.074×10^5
8.	Polydispersity index	2.347	2.287

Apart from the results stated earlier, the present theory has the merit of ability to predict the entire molecular weight distribution of the polymer product. The molecular weight distribution on weight basis for a particular set of the reaction conditions is plotted in Figure 8.
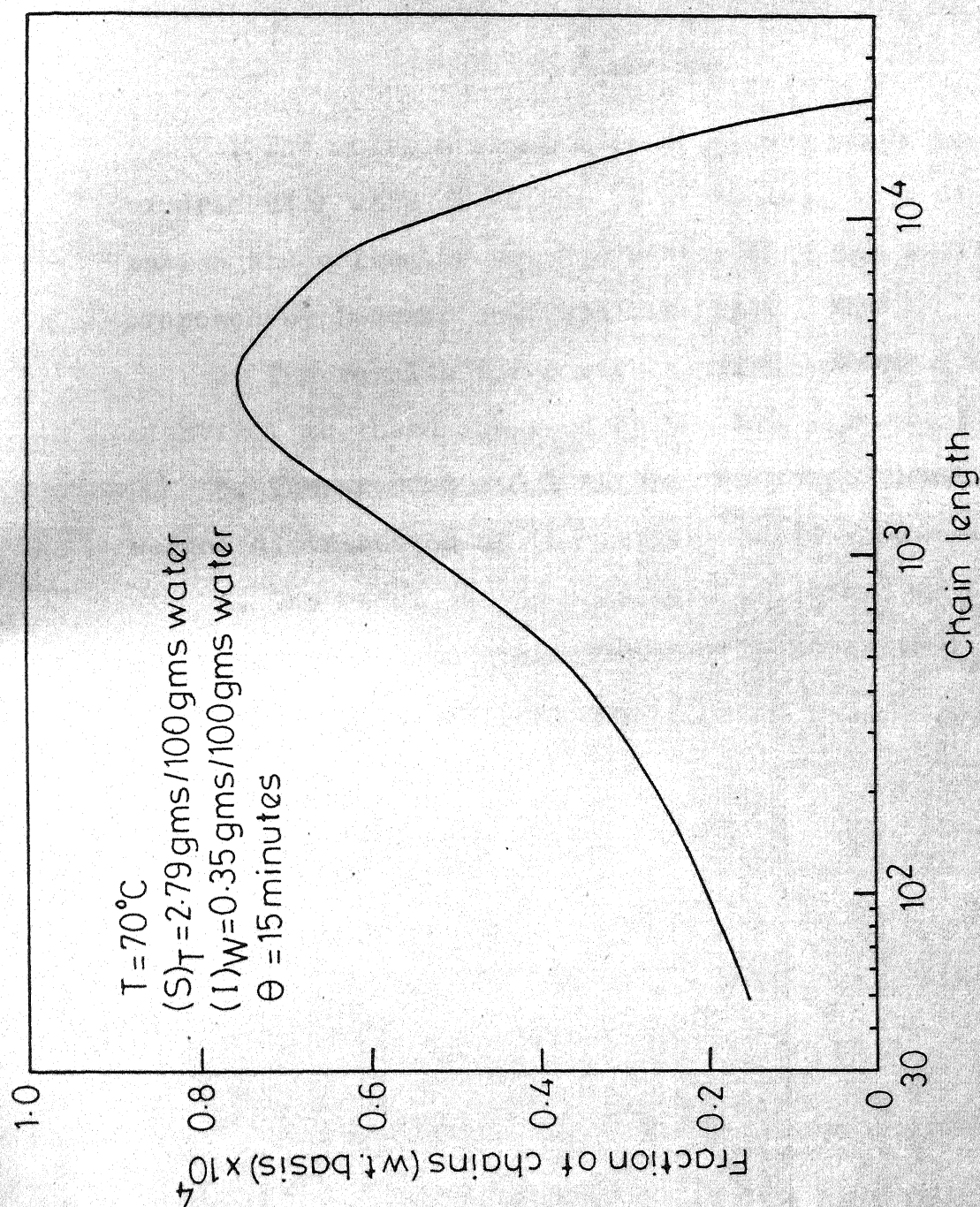
Fig. 8 - Molecular weight distribution of the polymer (wt. basis).

T = 70°C
$(S)_T$ = 2.79 gms/100 gms water
$(I)_W$ = 0.35 gms/100 gms water
θ = 15 minutes

Fraction of chains (wt. basis) × 10⁴

Chain length

75

CHAPTER 6

CONCLUSIONS

1. The model presented in this work predicts the experimental data on number of particles, rate of polymerization and molecular weights better than the earlier model proposed by DeGraff and Poehlein [21].

2. The results for particle size distribution are identical to those obtained by the RTD approach [21].

3. The present model allows the complete molecular weight distribution of the polymer product to evaluated.

4. The model can be made more effective by including the phenomena like chain transfer to impurities and termination by disproportionation.

REFERENCES

[1] W.V. Smith and R.H.Ewart, J.Chem. Phys., 16, 592 (1948).

[2] W.H. Stockmayer, J. Polym. Sci., 24, 314 (1957).

[3] J.T. O'Toole, J. Appl. Polym. Sci. 9, 1291 (1965).

[4] J.L. Gardon, J. Polym. Sci., Part A-1, 6, 623 (1968).

[5] J.L. Gardon, J. Polym.Sci., Part A-1, 6, 643 (1968).

[6] J.L. Gardon, J. Polym. Sci., Part A-1, 6,665 (1968).

[7] J.L. Gardon, J. Polym.Sci., Part A-1, 6, 687 (1968).

[8] J.L. Gardon, J. Polym. Sci., Part A-1, 6, 2853 (1968).

[9] J.L. Gardon, J. Polym. Sci., Part A-1, 6, 2859 (1968).

[10] J.L. Gardon, Br. Polym. J., 2, 1 (1970).

[11] J. Ugelstad and F.K. Hanson, J. Polym.Sci., 16, 1953 (1978).

[12] S.S. Medvedev, Proceedings of the International Symposium
     on Makromolecular Chemistry, Praugue, 1957, Pergamon,
     New York, 1958, p.174.

[13] K.W. Min and W.H.Ray, J. Appl. Polym. Sci., 22, 89 (1978).

[14] A.E. Alexander and D.H. Napper, Progress in Polymer
     Science (3) (A.D. Jenkins, ed.), Pergamon, 1975.

[15] J.W. Vanderhoff, Vinyl Polymerization, Part II (G.E. Ham,
     Ed.), Dekker, New York, 1969, Chap. 1.

[16] J.W. Vanderhoff, Advances in Emulsion Polymerization and
     Latex Technology, 8th Annual Short Course, Lehigh
     University (G. Poehlein, ed.), 1977.

[17] K.W. Min and W.H. Ray, J. Macro. Sci.-Rev. Macromol.
     Chem., C11, 177 (1974).

[18] D.B. Greshberg and J.F. Longfield, Unpublished Paper
presented at the Symp. Poly. Kinetics Cat. Systems,
54th AIChE Meeting, New York, 1961, Preprint No.10.

[19] T. Udea, S. Omi and H. Kubota, J.Chem. Eng. Japan, 2, 193
(1969); ibid, 4, 50(1971).

[20] M. Nomura, H. Kojima, M. Harada, V. Eguchi and S. Nagata,
J. Appl. Polym.Sci., 15, 675 (1971).

[21] A.W. DeGraff and G.W. Poehlein, J. Polym.Sci., A-2, 9, 1955
(1971).

[22] J.D.Stevens and J.O. Funderburk, I and E.C., Proc. Des.
and Dev., 11, 360 (1972).

[23] R.W. Thompson and J.W.Stevens, Chem.Eng. Sci., 32, 311 (1977).

[24] R.W. Thompson, D.A. Cauley, A.J. Giglio and J.D. Stevens,
Chem. Eng. Sci., 33, 383 (1978).

[25] D. Cauley, A.J. Giglio and R.W. Thompson, Chem.Eng.Sci., 33,
979 (1978).

[26] V.A. Kirilov and W.H. Ray, 33, 1499 (1978).

[27] M.R. Grancio and D.J. Williams, J. Polym.Sci., A-1, 8, 2617
(1970).

[28] D.H. Naper, J. Polym. Sci., A-1, 9, 2089 (1971).

[29] A.D. Randolph and M.A. Larson, Theory of Particular
Processes, Academic Press, New York, 1971.

[30] K.W. Min, J.Appl. Polym.Sci., 22, 589 (1978).

[31] Brown, Numerical Solution of Systems of Nonlinear
Algebraic Equations [G.D. Byrne and C.A. Hall, Ed.],
Academic Press, 1973.

[32] J.L. Kuster and J.H. Mize, Optimization Techniques with Fortran, McGraw-Hill, New York, 1973.

[33] G.S. Beveridge and R.S. Schechter, Optimization: Theory and Practice, McGraw-Hill, New York, 1970.

[34] J. Brandrup, E.H. Immergut, Polymer Handbook, John Wiley and Sons, Inc., 1966.

[35] Blackley, Emulsion Polymerization, Applied Science Publishers, 1975.

[36] F.A. Bovey, Emulsion Polymerization, Interscience Publishers, 1955.

[37] P.J. Flory, Principles of Polymer Chemistry, Cornell University Press, 1953.

```
0100  C**********************************************************************
0200  C        THIS PROGRAM IS USED TO SIMULATE A CONTINUOUS STIRRED TANK
0300  C        REACTOR FOR EMULSION  POLYMERIZATION,THE INPUT DATA FOR THE
0400  C        PROGRAM CONSISTS OF VARIOUS CONSTANTS,FEED CONDITIONS AND
0500  C        VARIOUS REACTOR PARAMETERS,THIS PROGRAM IS ABLE TO CALCULATE
0600  C        THE CONVERSION,RATE OF POLYMERIZATION,AVERAGE PARTICLE SIZE,
0700  C        NUMBER OF PARTICLES,PARTICLE SIZE DISTRIBUTION AND MOLECULAR
0800  C        WT,DISTIBUTION OF THE RESULTANT POLYMER LATEX,
0900  C              INITIALLY THE BOX'S OPTIMIZATION PROCEDURE IS USED TO
1000  C        CALCULATE INITIAL GUESSES FOR THE SOLUTION OF FIRST SUBGROUP,THE
1100  C        EQUATIONS OF FIRST SUBGROUP (I.E,MATERIAL BALANCE EQUATIONS,
1200  C        Particle size distribution and radical number distribution)are
1300  C        THEN SOLVED,THE EQUATIONS OF SECOND GROUP ARE THEN SOLVED TO
1400  C        OBTAIN MOMENTS OF LIVE POLYMER CHAIN LENGTH DISTR,THESE RESULTS
1500  C        ARE THEN USED TO SOLVE THIRD GROUP TO OBTAIN MOMENTS OF DEAD
1600  C        POLYMER CHAIN LENGTH DISTR,VARIOUS OUTPUT VARIABLES ARE THEN
1700  C        CALCULATED AND THE REQUIRED DISTRIBUTIONS ARE GENERATED,
1800  C              THE DETAILS OF THE METHOD OF SOLUTION OF THE MODELLING
1900  C        EQUATIONS IS GIVEN IN THESIS[PAGE          ],
2000  C
2100  C        NOMENCLATURE_____
2200  C        VARIABLES ==========
2300  C        XSTART(1,1)_____TOTAL SOAP CONCENTRATION
2400  C        XSTART(1,2)_____INITIATOR CONCENTRATION
2500  C        XSTART(1,3)_____TOTAL MONOMER CONCENTRATION
2600  C        XSTART(1,4)_____MICELLE CONCENTRATION
2700  C        XSTART(1,5)_____MONOMER DROPLETS CONCENTRATION
2800  C        XSTART(1,6)_____MONOMER VOLUME FRACTION IN
2900  C        XSTART(1,7)_____ZEROTH MOMENT W,R,T, VOLUME
3000  C        XSTART(1,8)_____FIRST MOMENT W,R,T, VOLUME
3100  C        XSTART(1,9)_____SECOND MOMENT W,R,T, VOLUME
3200  C        XSTART(1,10)_____THIRD MOMENT W,R,T, VOLUME
3300  C        XSTART(1,11)_____FOURTH MOMENT W,R,T, VOLUME
3400  C
3500  C        XRAD(I)_____RADICAL NUMBER DISTRIBUTION ;NUMBER OF
3600  C                            PARTICLES HAVING I RADICALS,
3700  C        XLIVE(K,I)_____MOMENTS OF LIVE CHAIN LENGTH DISTRIBUTION,
3800  C                            [K TH MOMENT ; I RADICALS IN PARTICLE,]
3900  C        XDEAD(K,I)_____MOMENTS OF DEAD CHAIN LENGTH DISTRIBUTION,
4000  C                            [K TH MOMENT ; I RADICALS IN PARTICLE,]
4100  C        XFD(I)_____FEED CONDITIONS FOR VARIABLES OF FIRST SUB=
4200  C                            GROUP,
4300  C        FFD(I)
4400  C        FDLIV(K,I)_____FEED CONDITIONS OF XLIVE(K,I)
4500  C        FDDED(K,I)_____FEED CONDITIONS OF XDEAD(K,I)
4600  C
4700  C        CONSTANTS_____
4800  C        AKP_____PROPATION RATE CONSTANT,
4900  C        AKD_____RATE CONSTANT FOR DISSOCIATION OF INITIATOR
```

```
5000   C                                          IN AQUEOUS PHASE.
5100   C            AKIP_____RATE CONSTANT FOR DISSOCIATION OF INITIATOR
5200   C                                          IN POLYMER PHASE.
5300   C            AKZERO_____DESORPTION RATE CONSTANT.
5400   C            AKTC_____RATE CONSTANT FOR TERMINATION BY COMBINATIO
5500   C            AKTD_____RATE CONSTANT FOR TERMINATION BY DISPROPOR=
5600   C                                          TIONATION.
5700   C            AKFT_____RATE CONSTANT FOR CHAIN TRANSFER TO T
5800   C                                          TRANSFER AGENT.
5900   C            AKFM_____RATE CONSTANT FOR CHAIN TRANSFER TO MONOMER
6000   C            DP_____DENSITY OF POLYMER.
6100   C            DM_____DENSITY OF MONOMER.
6200   C            VMOL_____MOLAR VOLUME OF MONOMER.
6300   C            SAI_____MONOMER=POLYMER INTERACTION PARAMETER.
6400   C            YETA_____SURFACE TENSION.
6500   C            AMW_____MOLECULAR WT. OF MONOMER.
6600   C            THETA_____MEAN RESIDENCE TIME.
6700   C            TEMP_____TEMPERATURE.
6800   C            VW_____WATER:EMULSION RATIO.
6900   C            AIOW_____(IO)W ;SEE PAGE 53
7000   C            AMZERO_____(M)0  ;SEE PAGE 53
7100   C            VD_____VOLUME OF DROPLETS.
7200   C            VM_____VOLUME OF MICELLES.
7300   C            CE_____AREA OCCUPIED BY SOAP.
7400   C            AMM_____MONOMER CONCENTRATION IN MICELLES.
7500   C            FEE_____EFFICIENCY OF DECOMPOSITION OF THE
7600   C                                          INITIATOR.
7700   C            SWC_____CRITICAL MICELLE CONCENTRATION.
7800   C
7900   C
8000   C            IPRIN_____PRINTING OPTION : IPRIN=1===INTERMIDIATE
8100   C                                          ITERATIONS ARE PRINTED ;IPRIN=0===ONLY
8200   C                                          DATA AND ANSWER FROM SIMULATION ARE PRINTE
8300   C            IPT_____PRINTING OPTION :IPT=0===CONCISE OUTPUT
8400   C                                          IPT=1===DETAILED OUTPUT[ ENTIRE DISTR.
8500   C                                          ARE  CALCULATED/PRINTED]
8600   C            MX_____INPUT UNIT NUMBER TO BE TYPED BY THE  USER
8700   C            NX_____RESULT UNIT NUMBER  TO BE TYPED BY THE USER
8800   C                                          NX=1,20,21,22,23,24====RESULT ON DISK
8900   C                                          NX=5====RESULT AT THE TERMINAL
9000   C
9100   C
9200   C***********************************************************************
9300         DOUBLE PRECISION XSTART(14,16),R(14,11),F(14),G(16),H(16)
9400         DOUBLE PRECISION XC(11),X(20),XTEMP(6),B(25,1),XRAD(0:20)
9500         DOUBLE PRECISION T(25,25)
9600         DOUBLE PRECISION FFD(0:20),FDLIV(0:5,20),FDDED(0:5,0:20),XFD(11)
9700         DOUBLE PRECISION C50,C51,C52,C53,C54,C556,C56,C7,CT1,C22,C33,C44
9800         DOUBLE PRECISION XLIVE(0:5,20),XDEAD(0:5,0:20),XCORD(0:50)
9900         DOUBLE PRECISION YCORD(0:50),GSMAL(0:5,20),XLIVET(0:5)
```

```
10000          DOUBLE PRECISION C79,C78,C65,YETA,XZER,DASHI,DASHI1,DASHI2
10100          DOUBLE PRECISION DIFF,ONETH,TWOTH,TRP,PHI,AM,AIP,SUM1,SUM2
10200          DOUBLE PRECISION BRACE1,BRACE2,BRACE3,SAI
10300          DOUBLE PRECISION THETR,DUMMY,THETVE,V,ANUMBR,AIW, XL1,PART1
10400          DOUBLE PRECISION ZCORD(0:50 ),XVOL(0:5),XDEADT(0:5),XCUM(0:50)
10500          DOUBLE PRECISION X01,X02,C,DAK,X66,X667,XMAX
10600          INTEGER GAMMA
10700          ACCEPT *,MX
10800          ACCEPT *,NX
10900    C     TO READ INPUT PARAMETERS FOR CONSTRAINED OPTIMIZATION
11000          READ(MX,*) NOPT,M,K,ITMAX,IPRIN
11100          READ(MX,*) ALPHA,BETA,DELTA,GAMMA,DASHI
11200          READ(MX,*) (XSTART(1,J),J=1,NOPT)
11300          DO 100 II=2,K
11400    100    READ(MX,*) (R(II,J),J=1,NOPT)
11500    C     TO READ INPUT PARAMETERS FOR LINEAR EQUATIONS AND
11600    C     CONVERGENCE CRITERION.
11700          READ(MX,*) MAXIT,NUMSIG,IPT,NUMEQ
11800          READ(MX,*) EPS,DELNON,PREC,ZETA
11900    C     TO READ INPUT CONSTANTS FOR THE REACTOR.
12000    C
12100    C        RATE CONSTANTS_____
12200          READ(MX,*) AKP,AKD,AKZERO,AKIP,AKTD,AKTC,AKFT,AKFM
12300    C     MONOMER-POLYMER CHARECTARISTICS___
12400          READ(MX,*) AMW,DP,DM,VMOL,SAI
12500    C     REACTOR CHARECTARISTICS___
12600          READ(MX,*) THETA,TEMP,VW
12700    C     OTHER CONSTANTS___
12800          READ(MX,*) AIOW,AMZERO,VD,VM,CE,AMM,FEE,SWC,YETA
12900    C     TO READ FEED CONDITIONS___
13000          READ(MX,*) (FFD(KI),KI=1,10)
13100          DO 181 KI=0,5
13200          READ(MX,*)(FDLIV(KI,IK),IK=1,10)
13300    181    CONTINUE
13400          DO 182 KI=0,5
13500          READ(MX,*) (FDDED(KI,IK),IK=0,10)
13600    182    CONTINUE
13700          READ(MX,*) (XFD(KI),KI=1,NOPT)
13800          ANA=6.02E+23
13900          RR=8.2764E+07
14000    C
14100          ONETH=1.0/3.0
14200          TWOTH=2./3.0
14300    C     CALCULATION OF DIMENSIONLESS CONSTANTS.
14400          BRACE1=(36.0*3.142)**ONETH
14500          BRACE2=VD**TWOTH
14600          BRACE3=VM**TWOTH
14700    C
14800          C2=AKIP*ANA*ANA*VM*VM*AIOW*DP/(AKP*DM)
14900          C3=AKZERO*ANA*DP*BRACE3*BRACE1/(AKP*DM)
```

```
15000          C4=(AKTC+AKTD)*DP/(2.0*AKP*DM)
15100          C50=AIOW*CE/(BRACE1*BRACE3*AMZERO*ANA)
15200          C51=(VW*SWC*CE)/(BRACE1*BRACE3*AMZERO*ANA)
15300          C52=BRACE2/BRACE3
15400          C53=ANA*VM*DP*AMZERO/(AIOW*AMW)
15500          C54=AIOW*VMOL/(ANA*VD*AMZERO)
15600          C55=VM /VD
15700          C556=AMM*VM*VMOL/VD
15800          C56=AKP*DM/(AKD*ANA*DP*VM)
15900          CT1=1.0
16000          C7=2.0*AKD*ANA*DP*AIOW*VM*FEE*VW/(AKP*DM*AMZERO)
16100          THETR=THETA*AKP*DM/(ANA*DP*VM)
16200          DUMMY=1.0E+03
16300          C403=AKTC*DP/(2.0*AKP*DM)
16400          C404=AKTD*DP/(2.0*AKP*DM)
16500          THETVE=1.0/THETR
16600          C63=AKFT*AIOW*ANA*DP*VM/(AKP*DM)
16700          C101=DP*VM*ANA/(DM*VMOL)
16800          C102=AKFM*ANA*DP*VM/(AKP*DM*VMOL)
16900          DUMMY2=1.0E-02
17000          C65=((4.0*3.142/(3.0*VM))**ONETH)*(2.*VMOL*YETA/(RR*TEMP))
17100          DUMMY3=1.0E-03
17200          DUMMY4=4.2985E+03
17300  C*********************************************************************
17400  C
17500  C      THE FOLLOWING LINES SHOULD BE INSERTED  FOR SECOND APPROACH.
17600  C      X01=0.01
17700  C      X02=0.99
17800  C      C=0.00001
17900  C      WRITE(NX,4189)
18000  C
18100  C*********************************************************************
18200
18300  C         PRINTING THE DATA___
18400            WRITE(NX,3185)
18500            WRITE(NX,2001)
18600            WRITE(NX,3185)
18700            WRITE(NX,4181)
18800            WRITE(NX,2002) NOPT,M,K,ITMAX,IPRIN
18900            WRITE(NX,2003) ALPHA,BETA,DELTA,GAMMA,DASHI
19000            WRITE(NX,2004)
19100            WRITE(NX,2005) ((J,XSTART(1,J)),J=1,NOPT)
19200            WRITE(NX,2006)
19300            WRITE(NX,2007) MAXIT,NUMSIG,IPT,NUMEQ
19400            WRITE(NX,2008) EPS,DELNON,PREC,ZETA
19500            WRITE(NX,2009) AKP,AKD,AKZERO,AKIP,AKTD,AKTC,AKFT,AKFM
19600            WRITE(NX,2010)
19700            WRITE(NX,2011) AMW,DP,DM,VMOL,SAI
19800            WRITE(NX,2012)
19900            WRITE(NX,2013) THETA,TEMP,VW
```

```
20000            WRITE(NX,2014)
20100            WRITE(NX,2015) AIOW,AMZERO,VD,VM,CE,AMM,FEE,SWC,YETA
20200            WRITE(NX,3015)
20300            WRITE(NX,3016)((KI,FFD(KI)),KI=1,10)
20400            WRITE(NX,2113)
20500            DO 281 KI=0,5
20600            WRITE(NX,2114) (KI,IK,FDLIV(KI,IK),IK=1,10)
20700     281    CONTINUE
20800            WRITE(NX,2115)
20900            DO 282 KI=0,5
21000            WRITE(NX,2116) (KI,IK,FDDED(KI,IK),IK=1,10)
21100     282    CONTINUE
21200            WRITE(NX,2487)
21300            WRITE(NX,2488) (KI,XFD(KI),KI=1,NOPT)
21400     C
21500     C
21600     2001   FORMAT(//10X,'DATA FOR THE PROBLEM'//)
21700     2002   FORMAT(//5X,'NOPT = ',I2,5X,'M = ',I2,5X,'K = ',I2,5X,
21800            1'ITMAX = ',I5,5X,'IPRIN = ',I2)
21900     2003   FORMAT(/5X,'ALPHA = ',F10.6,5X,'BETA = ',F10.6,5X,'DELTA = ',
22000            1F10.6//5X,'GAMMA = ',I2,13X,'DASHI = ',F10.6)
22100     2004   FORMAT(//5X,'INITIAL GUESSES_____' )
22200     2005   FORMAT(/2(5X,'XSTART(',I2,') = ',E15.8))
22300     2006   FORMAT(//5X,'INPUT PARAMETERS FOR LINEAR EQUATIONS AND
22400            1THE CONVERGENCE CRITERION')
22500     2007   FORMAT(/5X,'MAXIT = ',I5,5X,'NUMSIG = ',I2,5X,'IPT = ',
22600            1I2/5X,'NUMEQ = ',I2)
22700     2008   FORMAT(/5X,'EPS   = ',E15.8,3X,'DELNON = ',E15.8,3X,
22800            1'PREC = ',E15.8/5X,'ZETA = ',E15.8)
22900     2009   FORMAT(/5X,'RATE CONSTANTS_____'/5X,'KP  = ',E15.8,5X,
23000            1'KD  = ',E15.8,5X,'KZERO = ',E15.8/5X,'KIP = ',E15.8,5X,
23100            1'KTD = ',E15.8,5X,'KTC  = ',E15.8/5X,'KFT = ',E15.8,5X,
23200            1'KFM = ',E15.8 )
23300     2010   FORMAT(/5X,'MONOMER_POLYMER CHARECTARISTICS___')
23400     2011   FORMAT(/5X,'MW   = ',E15.8,5X,'DP = ',E15.8,5X,'DM = ',E15.8/
23500            15X,'VMOL = ',E15.8,5X,'SAI = ',E15.8)
23600     2012   FORMAT(/5X,'REACTOR CHARECTARISTICS___')
23700     2013   FORMAT(/5X,'THETA = ',E15.8,5X,'TEMP = ',E15.8,5X,
23800            1'VW = ',E15.8)
23900     2014   FORMAT(/5X,'OTHER CONSTANTS___')
24000     2015   FORMAT(/5X,'AIOW = ',E15.8,4X,'AMZERO = ',E15.8,4X,'VD = ',
24100            1E15.8/5X,'VM   = ',E15.8,4X,'CE = ',E15.8,4X,'AMM = ',
24200            1E15.8 /5X,'FEE  = ',E15.8,4X,'SWC = ',E15.8,4X,'YETA = ',E15.8)
24300     3015   FORMAT(/5X,'FEED CONDITIONS_____'///5X,'RADICAL NUMBER DISTRIBUTI
24400            1ON_____')
24500     3016   FORMAT(/2(5X,'FFD(',I2,') = ',F10.6))
24600     2113   FORMAT(//5X,'LIVE POLYMER CHAINS_____',/)
24700     2114   FORMAT(/3(4X,'FDLIV(',I2,1H,,I2,') =',F8.4))
24800     2115   FORMAT(//5X,'DEAD POLYMER CHAINS_____',/)
24900     2116   FORMAT(/3(4X,'FDDED(',I2,1H,,I2,') =',F8.4))
```

```
25000    2487       FORMAT(/5X,'FEED COND. FOR VARIABLES OF INITIAL SET')
25100    2488       FORMAT(/3(5X,I2,4X,E11.4))
25200    4181       FORMAT(/7X,'DATA FOR CONSTRAINED OPTIMIZATION_____')
25300    C***************************************************************
25400    C
25500    C          THE FOLLOWING LINE SHOULD BE INSERTED FOR SECOND APPROCH.
25600    C4189       FORMAT(5X,'OUTPUT OBTAINED BY EVALUATING PHI FROM EQUATION
25700    C          4.21)
25800    C
25900    C***************************************************************
26000    C
26100    C
26200               CALL CONSX(NOPT,M,K,ITMAX,ALPHA,BETA,GAMMA,DELTA,XSTART,R,F,IT,
26300              1IEV2,NX,G,H,XC,IPRIN,DASHI,ONETH,TWOTH,BRACE1,BRACE2,
26400              1BRACE3,C50,C51,C52,C53,C54,C55,C556,C56,CT1,C7,THETR,DUMMY,XFD,
26500              1C65,SAI)
26600               IF(IT-ITMAX) 20,20,30
26700    20         IF(IPRIN.EQ.0) GO TO 4501
26800               WRITE(NX,2117) F(IEV2)
26900    2117       FORMAT(//5X,'FINAL FUNCTION VALUE =',D18.10,//)
27000               DO 300 J=1,NOPT
27100               WRITE(NX,16)J,XSTART(IEV2,J)
27200    16         FORMAT(2X,'X('I2,') = 'E20.8)
27300    300        CONTINUE
27400    4501       GO TO 899
27500    30         WRITE(NX,17)ITMAX
27600    17         FORMAT(2X,'WARNING MAIN;THE NUMBER OF ITERATIONS HAS EXCEEDED'
27700              12X,I4)
27800    C
27900    C***************************************************************
28000    C
28100    899        DO 568 KI=1,NOPT
28200               X(KI)=XSTART(IEV2,KI)
28300    568        CONTINUE
28400    999        DASHI1=DASHI
28500               XZER=X(7)
28600               DO 167 KI=1,4
28700               XTEMP(KI)=X(KI+7)
28800    167        CONTINUE
28900               NNON=NOPT+1
29000               X(NNON)=CONFR(TWOTH,XTEMP,XZER)
29100               CALL NONLIN(NNON,X,NUMSIG,MAXIT,IPRIN,EPS,DASHI1,NX,DELNON,PREC,
29200              1ONETH,TWOTH,BRACE1,BRACE2,BRACE3,C50,C51,C52,C53,C54,C55,C556,
29300              1C56,C7,CT1,C65,DUMMY,THETR,SAI,XFD,X66)
29400               DO 1111 IK=1,NNON
29500               IF(X(IK).LT.0.0) X(IK)=0.0
29600    1111       CONTINUE
29700               V=X(8)*DUMMY4/X(7)
29800               AIW=X(2)
29900               AM=X(4)
```

```
30000             AIP=0.0
30100             ANUMBR=X(7)/DUMMY3
30200             F23=X(12)*(DUMMY4**TWOTH)/DUMMY3
30300             C22=C2*V*AIP
30400             C33=C3/(V**ONETH)
30500             C44=C4/V
30600             C78=C7*(V**TWOTH)*AIW/(AM+F23)
30700             C79=C7*AM*AIW/(AM+F23)
30800     C
30900     C*****************************************************************
31000     C
31100     C         INSERT THE FOLLOWING LINES IN PLACE OF THE ENCLOSED SET OF LINES
31200     C         GIVEN ABOVE FOR THE SECOND APPROACH.
31300     C
31400     C899     DO 568 KI=1,5
31500     C        X(KI)=XSTART(IEV2,KI)
31600     C568     CONTINUE
31700     C        X667=XSTART(IEV2,6)
31800     C        DO 569 KI=7,NOPT
31900     C        X(KI-1)=XSTART(IEV2,KI)
32000     C        XFD(KI-1)=XFD(KI)
32100     C569     CONTINUE
32200     C999     DASHI1=DASHI
32300     C        XZER=X(6)
32400     C        DO 167 KI=1,4
32500     C        XTEMP(KI)=X(KI+6)
32600     C167     CONTINUE
32700     C        DAK=CONFR(ONETH,XTEMP,XZER)
32800     C        CALL GOLD(X01,X02,C,X667,DAK,XZER,SAI,C65,DUMMY4,ONETH)
32900     C        NNON=NOPT
33000     C        X(NNON)=CONFR(TWOTH,XTEMP,XZER)
33100     C998     X66=X667
33200     C        CALL NONLIN(NNON,X,NUMSIG,MAXIT,IPRIN,EPS,DASHI1,NX,DELNON,PREC,
33300     C        1ONETH,TWOTH,BRACE1,BRACE2,BRACE3,C50,C51,C52,C53,C54,C55,C556,
33400     C        1C56,C7,CT1,C65,DUMMY,THETR,SAI,XFD,X66)
33500     C        DO 1111 KI=1,NNON
33600     C        IF(X(KI).LT.0.0) X(KI)=0.0
33700     C1111    CONTINUE
33800     C        XZER=X(6)
33900     C        DO 168 KI=1,4
34000     C        XTEMP(KI)=X(KI+6)
34100     C168     CONTINUE
34200     C        DAK=CONFR(ONETH,XTEMP,XZER)
34300     C        CALL GOLD(X01,X02,C,XMAX,DAK,XZER,SAI,C65,DUMMY4,ONETH)
34400     C        X667=XMAX
34500     C        DIFF=X667-X66
34600     C          IF(IPRIN.EQ.0) GO TO 4938
34700     C        WRITE(NX,4086) DIFF,X667
34800     C4086    FORMAT(/5X,'DIFFERANCE IN X66 = ',D15.8/5X,'X66 = ',D15.8/)
34900     C4938    IF(ABS(DIFF).LT.0.0001) GO TO 987
```

```
35000    C          GO TO 998
35100    C987        V=X(7)*DUMMY4/X(6)
35200    C          AIW=X(2)
35300    C          AM=X(4)
35400    C          AIP=0.0
35500    C          ANUMBR=X(6)/DUMMY3
35600    C          F23=X(11)*(DUMMY4**TWOTH)/DUMMY3
35700    C          C22=C2*V*IP
35800    C          C33=C3/(V**ONETH)
35900    C          C44=C4/V
36000    C          C78=C7*(V**TWOTH)*AIW/(AM+F23)
36100    C          C79=C7*AM*AIW/(AM+F23)
36200    C
36300    C**********************************************************************
36400               CALL LIN(NUMEQ,FFD,ANUMBR,THETR,C22,C33,C78,C79,C44,NX,DASHI2,
36500              1NTEMP3,B,IPRIN)
36600               DIFF=DASHI2-DASHI1
36700               IF(IPRIN.EQ.0) GO TO 4502
36800               WRITE(NX,4002) DIFF
36900    4002       FORMAT(/2X,'DIFF    ',E15.8/)
37000    4502       IF(ABS(DIFF).LT.ZETA) GO TO 98
37100               DASHI=DASHI2
37200               GO TO 999
37300    98         IF(IPRIN.EQ.0) GO TO 4503
37400               WRITE(NX,2017)
37500    2017       FORMAT(/' FUNCTION VALUES AFTER FIRST SET')
37600               WRITE(NX,2018)((I,X(I)),I=1,NNON)
37700    2018       FORMAT(1(5X,'X(',I2,') = ',E15.8))
37800    4503       DO 187 IK=1,NTEMP3
37900               KI=IK
38000               XRAD(KI-1)=B(KI,1)
38100    187        CONTINUE
38200               IF(IPRIN.EQ.0) GO TO 4504
38300               WRITE(NX,2019)((KI,XRAD(KI)),KI=0,NTEMP3)
38400    2019       FORMAT(/,1(5X'XRAD(',I2,') = ',E15.8))
38500    C          LIVE POLYMER CHAIN LENGTH DISTRIBUTION STARTS.
38600    4504       NTEMP2=NTEMP3-1
38700               PHI=X(6)
38800    C
38900    C**********************************************************************
39000    C
39100    C          INSERT THE FOLLOWING LINE INSTEAD OF THE PREVIOUS LINE FOR
39200    C          SECOND APPROACH.
39300    C          PHI=X66
39400    C
39500    C**********************************************************************
39600    C
39700               DO 878 KI=1,NTEMP2
39800               XLIVE(0,KI)=KI*XRAD(KI)
39900    878        CONTINUE
```

```
40000              DO 857 MONT=1,4
40100              MOMENT=MONT
40200              CALL SET2(NTEMP3,C22,C33,C44,C78,C79,C63,C102,C101,TRP,
40300              1MOMENT,PHI,THETVE,FDLIV,XRAD,XLIVE,T,B)
40400              M=1
40500              CALL MATIN(T,NTEMP2,B,M,DETER)
40600              DO 857 KI=1,NTEMP2
40700              XLIVE(MOMENT,KI)=B(KI,1)
40800      857     CONTINUE
40900              DO 862 MONT=0,4
41000              IF(IPRIN.EQ.0) GO TO 4505
41100              WRITE(NX,4250)
41200      4250    FORMAT(/5X,'LIVE POLYMER CHAIN LENGTH DISTRIBUTION_____'/)
41300              WRITE(NX,2499) MONT
41400      2499    FORMAT(/5X,'MOMENT  =',I2)
41500              WRITE(NX,2500)(KI,XLIVE(MONT,KI),KI=1,NTEMP2)
41600      2500    FORMAT(2(5X,I2,5X,E15.8))
41700      862     CONTINUE
41800      C       DEAD POLYMER CHAIN LENGTH DISTRIBUTION STARTS.
41900      4505    DO 911 MOMENT=0,4
42000              DO 911 IRAD=1,NTEMP3
42100              GSMAL(MOMENT,IRAD)=XLIVE(MOMENT,IRAD)/XRAD(IRAD)
42200      911     CONTINUE
42300              DO 912 MONT=0,4
42400              MOMENT=MONT
42500              CALL SET3(NTEMP3,C22,C33,C78,C44,C102,C63,C403,C404,V,
42600              1TRP,PHI,THETVE,XRAD,FDDED,XLIVE,GSMAL,MOMENT,T,B)
42700              M=1
42800              CALL MATIN(T,NTEMP3,B,M,DETER)
42900              DO 912 KI=1,NTEMP3
43000              XDEAD(MOMENT,KI-1)=B(KI,1)
43100      912     CONTINUE
43200      C
43300      C
43400              IF(IPRIN.EQ.0) GO TO 4506
43500              WRITE(NX,4251)
43600      4251    FORMAT(/5X,'DEAD POLYMER CHAIN LENGTH DISTRIBUTION_____'/)
43700              DO 963 MONT=0,4
43800              WRITE(NX,2499) MONT
43900              WRITE(NX,2500) (KI,XDEAD(MONT,KI),KI=0,NTEMP2)
44000      963     CONTINUE
44100      4506    DO 872 MOMENT=0,4
44200              SUM1=0.0
44300              DO 871 KI=1,NTEMP2
44400              SUM1=SUM1+XLIVE(MOMENT,KI)
44500      871     CONTINUE
44600              XLIVET(MOMENT)=SUM1
44700      872     CONTINUE
44800              DO 874 MOMENT=0,4
44900              SUM2=0.0
```

```
45000          DO 873 KI=0,NTEMP2
45100          SUM2=SUM2+XDEAD(MOMENT,KI)
45200   873    CONTINUE
45300          XDEADT(MOMENT)=SUM2
45400          TYPE *,SUM2
45500   874    CONTINUE
45600          AMNWT=AMW*XDEADT(1)/XDEADT(0)
45700
45800          AMWWT= AMW*XDEADT(2)/XDEADT(1)
45900          PDI=AMWWT/AMNWT
46000          VAVE=X(8)*DUMMY4*VM/X(7)
46100   C
46200   C*****************************************************************
46300   C
46400          DAVE2=((VAVE*6.0/3.142)**ONETH)*(1.0D+08)
46500          DAVE1=(CONFR(ONETH,XTEMP,XZER)*(DUMMY4**ONETH)*((6.0*VM/3.142)
46600         1**ONETH)/X(7))*1.00D+08
46700          PATNUM=X(7)*AMZERO*ANA/DUMMY3
46800   C
46900   C*****************************************************************
47000   C
47100   C      INSERT THE FOLLOWING LINES IN PLACE OF THE SET OF LINES
47200   C      ENCLOSED ABOVE  FOR THE SECOND APPROACH.
47300   C      VAVE=X(7)*DUMMY4*VM/X(6)
47400   C      DAVE2=((VAVE*6.0/3.142)**ONETH)*(1.0D+08)
47500   C      DAVE1=(CONFR(ONETH,XTEMP,XZER)*(DUMMY4**ONETH)*((6.0*VM/3.142)
47600   C     1**ONETH)/X(6))*1.00D+08
47700   C      PATNUM=X(6)*AMZERO*ANA/DUMMY3
47800   C
47900   CC****************************************************************
48000   C
48100          CONVER=(XFD(3)-X(3))/XFD(3)
48200          RATE=(XFD(3)-X(3))*AIOW*3.6E+06/(THETA*DUMMY2)
48300   C
48400   C
48500   C
48600   C
48700   C      OUTPUT SECTION BEGINS_____
48800          WRITE(NX,3186)
48900   3186   FORMAT(1H1)
49000          WRITE(NX,3185)
49100   3185   FORMAT(1X,'****************************************************
49200         1*********************')
49300          WRITE(NX,2147)
49400   2147   FORMAT(/10X,'OUTPUT FROM THE PROGRAM_____'/)
49500          WRITE(NX,3185)
49600          WRITE(NX,2148) PATNUM,VAVE,DAVE2,DAVE1,CONVER,RATE
49700   2148   FORMAT(/5X,'(1)   NUMBER OF PARTICLES (/CC EMULSION)____',E15.8/
49800         15X,'(2)   AVERAGE VOLUME OF PARTICLES (CC)____',E15.8/
49900         15X,'(3)   AVERAGE DIA ;VOLUME BASIS ("A)____',E15.8/
```

```
50000              15X,'(4)   AVERAGE DIA ;NUMBER BASIS ("A)_____',E15.8//
50100              15X,'(5)   CONVERSION OBTAINED_____',E15.8/
50200              15X,'(6)   RATE (MOLES/(LIT.HOUR))_____',E15.8/)
50300              WRITE(NX,2149) AMNWT,AMWWT,PD1
50400      2149    FORMAT(5X,'(7)   NUMBER AVERAGE MOLECULAR WEIGHT_____',E15.8/
50500              15X,'(8)   WEIGHT AVERAGE MOLECULAR WEIGHT_____',E15.8/
50600              15X,'(9)   POLYDISPERSITY INDEX_____',E15.8//)
50700              WRITE(NX,3185)
50800              IF(IPT.EQ.0) GO TO 4510
50900              WRITE(NX,3186)
51000              WRITE(NX,3187)
51100      3187    FORMAT( /5X,'DETAILS OF THE DISTRIBUTIONS FOLLOW_____'/)
51200              WRITE(NX,3185)
51300    C
51400    C
51500              IWHICH=0
51600              XL1=1.0
51700              PART1=5000.0
51800              CALL DISTR(XL1,PART1,XLIVET,IWHICH,XCORD,YCORD,ZCORD,XCUM)
51900              WRITE(NX,2150)
52000      2150    FORMAT(/5X,'LIVE POLYMER CHAIN LENGTH DISTRIBUTION'/)
52100              WRITE(NX,3185)
52200              WRITE(NX,2151)
52300      2151    FORMAT(/5X,'NO',5X,'MOL. WT.',5X,'NUMB.BASIS',5X,'WT.BASIS
52400             1(NORMAL)'/)
52500              SUM1=0.0
52600              PART1=XCORD(2)-XCORD(1)
52700              DO 870 KI=0,50
52800              SUM1=SUM1+ZCORD(KI)*PART1
52900      870     CONTINUE
53000              DO 863 KI=0,50
53100              ZCORD(KI)=ZCORD(KI)/SUM1
53200      863     CONTINUE
53300              DO 879 KI=0,50
53400              WRITE(NX,2152) KI,XCORD(KI),YCORD(KI),ZCORD(KI)
53500      2152    FORMAT(5X,I2,3X,E11.4,3X,E11.4,3X,E11.4)
53600      879     CONTINUE
53700    C
53800    C
53900    C
54000              CALL DISTR(XL1,PART1,XDEADT,IWHICH,XCORD,YCORD,ZCORD,XCUM)
54100              SUM1=0.0
54200              PART1=XCORD(2)-XCORD(1)
54300              DO 4851 KI=0,50
54400              SUM1=SUM1+ZCORD(KI)*PART1
54500      4851    CONTINUE
54600              DO 4853 KI=0,50
54700              ZCORD(KI)=ZCORD(KI)/SUM1
54800      4853    CONTINUE
54900              WRITE(NX,3186)
```

```
55000            WRITE(NX,3188)
55100            WRITE(NX,3185)
55200    3188    FORMAT( //)
55300            WRITE(NX,2153)
55400    2153    FORMAT(/5X,'DEAD POLYMER CHAIN LENGTH DISTRIBUTION'/)
55500            WRITE(NX,3185)
55600            WRITE(NX,2154)
55700    2154    FORMAT(/5X,'NO',5X,'MOL.WT.',6X,'NUMB.BASIS',5X,'WT.BASIS
55800           1(NORMAL)'/)
55900            DO 891 KI=0,50
56000            WRITE(NX,2152) KI,XCORD(KI),YCORD(KI),ZCORD(KI)
56100    891     CONTINUE
56200    C
56300    C
56400            XL1=1.0
56500            PART1=100.0
56600            IWHICH=1
56700            DO 892 KI=0,4
56800            XVOL(KI)=X(KI+7)*(DUMMY4**KI)/DUMMY3
56900    892      CONTINUE
57000            CALL DISTR(XL1,PART1,XVOL,IWHICH,XCORD,YCORD,ZCORD,XCUM)
57100            DO 2176 KI=0,50
57200            XCORD(KI)=((XCORD(KI)*VM*6.0/3.142)**ONETH)*(1.0D+08)
57300    2176    CONTINUE
57400            WRITE(NX,3186)
57500            WRITE(NX,3188)
57600            WRITE(NX,3185)
57700            WRITE(NX,2155)
57800    2155    FORMAT(/5X,'PARTICLE SIZE DISTRIBUTION [CUMULATIVE]'/)
57900            WRITE(NX,3185)
58000            WRITE(NX,2156)
58100    2156    FORMAT(/5X,'NO',7X,'DIA("A)',9X,'FRACTION' /)
58200            DO 890 KI=0,50
58300            WRITE(NX,2157) KI,XCORD(KI),XCUM(KI)
58400    890     CONTINUE
58500    2157    FORMAT(5X,I2,5X,E11.4,6X,E11.4)
58600    C
58700    4510    WRITE(NX,3186)
58800            STOP
58900            END
59000    C
59100    C
59200            SUBROUTINE NONLIN(N,X,NUMSIG,MAXIT,IPRIN,EPS,DASHI,NX,DELNON,
59300           1PREC,ONETH,TWOTH,BRACE1,BRACE2,BRACE3,C50,C51,C52,C53,C54,C55,
59400           1C556,C56,C7,CT1,C65,DUMMY,THETR,SAI,XFD,X66)
59500    C*****************************************************************
59600    C
59700    C          THIS SUBROUTINE SOLVES A SET OF NONLINEAR ALGEBRAIC
59800    C      EQUATIONS.IT IS USED TO SOLVE THE EQUATIONS OF FIRST SUBGROUP
59900    C      OF GROUP 1.OTHER SUBROUTINES DIRECTLY CONNECTED WITH THIS
```

```
60000   C          SUBROUTINE ARE BACK,AMAX1,AMINÍ,FUNX.
60100   C                   SEE REF 31  FOR DETAILS.
60200   C
60300   C          DISCRIPTION OF PRINCIPAL ARGUMENTS____
60400   C          N_____NUMBER OF EQUATIONS.
60500   C          X(I)_____THE VECTOR OF VARIABLES.
60600   C          NUMSIG_____CONVERGENCE CRITERION.
60700   C          EPS_____CONVERGENCE CRITERION.
60800   C          MAXIT_____MAXIMUM NUMBER OF ITERATIONS ALLOWED BY
60900   C                                       THIS PROCEDURE.
61000   C          PREC_____MINIMUM ALLOWABLE STEP SIZE FOR CALCULATING
61100   C                                       PARTIAL DERIVATIVES.
61200   C          DELNON_____MINIMUM ALLOWABLE ABSOLUTE VALUE OF PARTIAL
61300   C                                       DERIVATIVES.
61400   C
61500   C*********************************************************************************
61600              DOUBLE PRECISION X(12),PART(12),TEMP(12),COE(12,13),XFD(11)
61700              DOUBLE PRECISION FMAX,FPLUS,F,ABSFF,HOLD,FACTOR,ETA,H
61800              DOUBLE PRECISION ABB,DERMAX,TEST,AA,CC,X66,DASHI,ONETH,TWOTH
61900              DOUBLE PRECISION BRACE1,BRACE2,BRACE3,THETR,DUMMY,SAI
62000              DOUBLE PRECISION  C50,C51,C52,C53,C54,C55,C556,C56,C7,CT1,C65
62100              DIMENSION ISUB(12),LOKP(12,12)
62200              RELCON=10.0E+00**(-NUMSIG)
62300              JTEST=1
62400              DO 700 M=1,MAXIT
62500              DO 1111 KI=1,N
62600              IK=KI
62700              IF(X(IK).LE.0.0) X(IK)=0.0
62800   1111       CONTINUE
62900              LQUIT=0
63000              FMAX=0.0
63100              M1=M-1
63200              IF(1PRIN.NE.1) GO TO 9
63300              WRITE(NX,49)M1,(X(I),I=1,N)
63400   49         FORMAT(5X,'M1= ',I5/5(5X,E11.4))
63500              NW=1
63600   9          DO 10 J=1,N
63700   10         LOKP(1,J)=J
63800              DO 5555 K=1,N
63900              IF(K-1) 134,134,131
64000   131        KMIN=K-1
64100              KTEMP=K
64200              CALL BACK (KMIN,N,X,ISUB,COE,LOKP,KTEMP)
64300
64400   134        JJJ=K
64500              CALL FUNX(X,F,JJJ,DASHI,
64600             1ONETH,TWOTH,BRACE1,BRACE2,BRACE3,C50,C51,C52,C53,C54,C55,C556,
64700             1C56,C7,CT1,C65,DUMMY,THETR,SAI,XFD,X66)
64800              ABSFF=ABS(F)
64900              FMAX=AMAX1(FMAX,ABSFF)
```

```
65000      C
65100      C
                   IF(ABS(F).GE.EPS) GO TO 1345
65300              LQUIT=LQUIT+1
65400              IF(LQUIT.NE.N) GO TO 1345
65500              GO TO 725
65600      1345    FACTOR=0.001E+00
65700      135     ITALLY=0
65800              DO 200 I=K,N
65900              ITEMP=LOKP(K,I)
66000              HOLD=X(ITEMP)
66100              ETA=FACTOR*ABS(HOLD)
66200              H=AMIN1(FMAX,ETA)
66300      CCC     H=ETA
66400              IF(H.LT.PREC) H=PREC
66500              X(ITEMP)=HOLD+H
66600              IF(K-1) 161,161,151
66700              KTEMP=K
66800      151     CALL BACK(KMIN,N,X,ISUB,COE,LOKP,KTEMP)
66900      161     JJJ=K
67000              CALL FUNX(X,FPLUS,JJJ,DASHI,
67100             1ONETH,TWOTH,BRACE1,BRACE2,BRACE3,C50,C51,C52,C53,C54,C55,C556,
67200             1C56,C7,CT1,C65,DUMMY,THETR,SAI,XFD,X66)
67300              PART(ITEMP)=(FPLUS-F)/H
67400              X(ITEMP)=HOLD
67500              IF(ABS(PART(ITEMP)).LT.DELTA) GO TO 190
67600      C
67700              ABB=ABS(F/PART(ITEMP))
67800              IF(ABB.LE.1.E+15) GO TO 200
67900
68000      190     ITALLY=ITALLY+1
68100      200     CONTINUE
68200      C
68300              NK=N-K
68400              IF(ITALLY.LE.NK) GO TO 202
68500      C
68600              FACTOR=FACTOR*10.0E+00
68700              IF(FACTOR.GT.11.) GO TO 775
68800              GO TO 135
68900      202     IF(K.LT.N) GO TO 203
69000              IF(ABS(PART(ITEMP)).LT.DELTA) GO TO 775
69100              NPL1=N+1
69200              COE(K,NPL1)=0.0
69300              KMAX=ITEMP
69400              GO TO 500
69500      203     KMAX=LOKP(K,K)
69600              DERMAX=ABS(PART(KMAX))
69700              KPLUS=K+1
69800              DO 210 I=KPLUS,N
69900              JSUB=LOKP(K,I)
```

```
70000          TEST=ABS(PART(JSUB))
70100          IF(TEST.LT.DERMAX) GO  TO 209
70200          DERMAX=TEST
70300          LOKP(KPLUS,I)=KMAX
70400          KMAX=JSUB
70500          GO TO 210
70600   209    LOKP(KPLUS,I)=JSUB
70700   210    CONTINUE
70800          IF(ABS(PART(KMAX)).EQ.0.0) GO TO 775
70900          ISUB(K)=KMAX
71000          COE (K,N+1)=0.0
71100          DO 220 J=KPLUS,N
71200          JSUB=LOKP(KPLUS,J)
71300          COE(K,JSUB)=-PART(JSUB)/PART(KMAX)
71400   C
71500          NPL1=N+1
71600          COE(K,NPL1)=COE(K,NPL1)+PART(JSUB)*X(JSUB)
71700   220    CONTINUE
71800          NPL1=N+1
71900   500    COE(K,NPL1)=(COE(K,NPL1)-F)/PART(KMAX)+X(KMAX)
72000   C      WRITE(NX,6666) K,(ISUB(IK),IK=1,12),((LOKP(IJ,JI),IJ=1,12),
72100   C     1JI=1,12)
72200   C6666  FORMAT(I2/5(2X,E15.7)/5(2X,E15.7))
72300   5555   CONTINUE
72400          NPL1=N+1
72500          X(KMAX)=COE(N,NPL1)
72600          IF(N.EQ.1) GO TO 610
72700          N1=N-1
72800          CALL BACK(N1,N,X,ISUB,COE,LOKP,K)
72900   610    IF(M-1) 650,650,625
73000   C
73100   625    DO 630 I=1,N
73200          AA=ABS(TEMP(I)-X(I))
73300          CC=ABS(X(I))*RELCON
73400
73500          IF(AA.GT.CC) GO TO 649
73600   630    CONTINUE
73700   C
73800   C
73900          JTEST=JTEST+1
74000          IF(JTEST-3) 650,725,725
74100   649    JTEST=1
74200   650    DO 660 I=1,N
74300   660    TEMP(I)=X(I)
74400   700    CONTINUE
74500          WRITE(NX, 1753)
74600   1753   FORMAT(/2X,'WARNING NONLIN;NO CONVERGENCE MAXIMUM NO OF
74700          1ITERATION USED')
74800          IF(IPRIN.NE.1) GO TO 800
74900          WRITE(NX, 1763)
```

```
75000   1763      FORMAT(/'FUNCTION VALUE AT LAST APROXIMATION FOLLOW')
75100             IFLAG=1
75200             GO TO 7777
75300   725       IF(IPRIN.NE.1) GO TO 800
75400             IFLAG=0
75500   7777      DO 750 IK=1,N
75600             KII=IK
75700
75800             CALL FUNX(X,PART(KII),KII,DASHI,
75900            1ONETH,TWOTH,BRACE1,BRACE2,BRACE3,C50,C51,C52,C53,C54,C55,C556,
76000            1C56,C7,CT1,C65,DUMMY,THETR,SAI,XFD,X66)
76100   750       CONTINUE
76200             IF(IFLAG.NE.1) GO TO 8777
76300             WRITE(NX, 7788)(PART(K),K=1,N)
76400   7788      FORMAT(3F15.8)
76500             GO TO 800
76600   8777      WRITE(NX, 751)
76700   751       FORMAT(//5X,'CONVERGENCE HAS BEEN ACHIVED .THE FUNCTI')
76800             WRITE(NX, 7515)(PART(K),K=1,N)
76900   7515      FORMAT(5X,'VALUES AT FINAL APPROXIMATION FOLLOW'//
77000            1(2X,F16.8,3(2X,F16.8)))
77100             WRITE(NX, 8888)(X(KI),KI=1,N)
77200   8888      FORMAT(5X,'ANSWER '/3(2X,F16.8))
77300             GO TO 800
77400   775       WRITE(NX,752)
77500   752       FORMAT(//5X,'WARNING NONLIN;MODIFIED JACOBIAN IS SINGULAR TRY ')
77600             WRITE(NX,7525)
77700   7525      FORMAT(5X,'DIFFERENT INITIAL APPROXIMATIOS ')
77800             DO 4005 I=1,N
77900             TYPE 4500,(COE(I,J),J=1,N)
78000   4005      CONTINUE
78100             TYPE *,(ISUB(I),I=1,N)
78200   4500      FORMAT(/,3X,6(2X,E11.4))
78300   C         MAXIT=M+1
78400   800       RETURN
78500             END
78600   C
78700   C
78800             SUBROUTINE BACK(KMIN,N,X,ISUB,COE,LOKP,K)
78900   C***************************************************************************
79000   C
79100   C                THIS SUBROUTINE IS CALLED FROM NONLIN.IT IS USED FOR
79200   C         BACKSOLVING THE SETS OF EQUATIONS OBTAINED IN NONLIN  AT VARIOUS
79300   C         STAGES  OF ITERATION.
79400   C
79500   C***************************************************************************
79600             DOUBLE PRECISION X(12),COE(12,13)
79700             DIMENSION ISUB(12),LOKP(12,12)
79800             DO 200  KK=1,KMIN
79900             KM=KMIN-KK+2
```

```
80000    C
80100            KM1=KM-1
80200            KMAX=ISUB(KM1)
80300    C
80400            X(KMAX)=0.0
80500            DO 100 J=KM,N
80600            JSUB=LOKP(KM,J)
80700            KM1=KM-1
80800            X(KMAX)=X(KMAX)+COE(KM1,JSUB)*X(JSUB)
80900    100     CONTINUE
81000            KM1=KM-1
81100            NPL1=N+1
81200            X(KMAX)=X(KMAX)+COE(KM1,NPL1)
81300    200     CONTINUE
81400            RETURN
81500            END
81600    C
81700    C
81800            FUNCTION AMAX1(A,C)
81900    C*****************************************************************
82000    C
82100    C                 THIS SUBROUTINE IS CALLED FROM NONLIN.IT REPORTS THE
82200    C        GREATER OF THE TWO NUMBERS REPRESENTING THE ARGUMENTS(I.E.A AND
82300    C        C).
82400    C
82500    C*****************************************************************
82600            DOUBLE PRECISION A,C
82700            IF(A.GT.C) C=A
82800            AMAX1=C
82900            RETURN
83000            END
83100    C
83200    C
83300            FUNCTION AMIN1(P,Q)
83400    C*****************************************************************
83500    C
83600    C                 THIS SUBROUTINE IS CALLED FROM NONLIN.IT REPORTS THE
83700    C        SMALLER OF THE TWO NUMBERS REPRESENTING THE ARGUMENTS(I.E.P AND
83800    C        Q).
83900    C
84000    C*****************************************************************
84100            DOUBLE PRECISION P,Q
84200            IF(P.GT.Q) P=Q
84300            AMIN1=P
84400            RETURN
84500            END
84600    C
84700    C
84800            SUBROUTINE FUNX(X,RES,K,DASHI,
84900           1ONETH,TWOTH,BRACE1,BRACE2,BRACE3,C50,C51,C52,C53,C54,C55,C556,
```

```
85000          1C56,C7,CT1,C65, DUMMY,THETR,SAI,XFD)
85100   C*********************************************************************
85200   C
85300   C                 THIS SUBROUTINE IS CALLED FROM NONLIN.IT REPORTS THE
85400   C          VALUE OF K TH EQUATION OF THE SET OF ALGEBRAIC EQUATIONS   TO
85500   C          BE SOLVED   BY NONLIN.
85600   C
85700   C*********************************************************************
85800          DOUBLE PRECISION X(12),XFD(12),F(6),FFD(6),FOLD(6)
85900          DOUBLE PRECISION RES,DASHI,ONETH,TWOTH
86000          DOUBLE PRECISION BRACE1,BRACE2,BRACE3,DUMMY,THETR,SAI
86100          DOUBLE PRECISION C50,C51,C52,C53,C54,C55,C556,C56,C7,CT1,C65
86200   C      DOUBLE PRECISION FZERO,FFDZER
86300   C      COMMON ONETH,TWOTH,BRACE1,BRACE2,BRACE3,C50,C51,C52,C53,C54,C55,
86400   C      1C556,C56,CT1,C7,THETR,DUMMY
86500          NMAX=4
86600   C
86700   C
86800   C
86900          W1=8.08E-04
87000          W2=47.69
87100          W3=3.807E-02
87200          W4=2.937E+06
87300          W5=7.0998E-01
87400          W6=0.63
87500          W7=0.1981
87600          FOLD(1)=8.5111E+02
87700          FOLD(2)=7.3E+06
87800          FOLD(3)=9.432E+10
87900          FOLD(4)=1.622E+15
88000   C
88100          DUMMY1=1.0E+06
88200          DUMMY2=1.0E-02
88300          DUMMY3=1.0E-03
88400          DUMMY4=4.2985E+03
88500   C
88600   C      X ARE THE OUTPUT VARIABLES FROM THE REACTOR
88700   C      XFD ARE THE FEED CHARECTARISTIC OF AFORESAID VARIABLES
88800          N=K=8
88900          FZERO=X(7)
89000          FFDZER=XFD(7)
89100          DO 210 INTEMP=1,NMAX
89200          NTEM=INTEMP
89300          F(NTEM)=X(NTEM+7)
89400          FFD(NTEM)=XFD(NTEM+7)
89500   210    CONTINUE
89600          IF(K.GT.7) GO TO 205
89700          GO TO (1,2,3,4,5,6,7),K
89800   1      CONH=CONFR(TWOTH,F,FZERO)
89900          RES=X(12)-CONH
```

```
90000              RETURN
90100    2         RES=(XFD(1)-X(1))/THETR
90200              RES=RES/W1
90300              RETURN
90400    3         RES=C56*(XFD(2)-X(2))/THETR-X(2)
90500              RES=RES/W2
90600              RETURN
90700    4         RES=(XFD(3)-X(3))/(THETR*DUMMY2)-C53*X(6)*(FZERO/DUMMY3)*DASHI
90800              RES=RES/W3
90900              RETURN
91000    5         RES=C50*X(1)-X(4)-C51-X(12)*(DUMMY4**TWOTH)/
91100             1DUMMY3-C52*X(5)
91200              RES=RES/W4
91300              RETURN
91400    6         RES=C54*X(3)/DUMMY2-(X(5))-C55*(F(1)*DUMMY4/DUMMY3)*
91500             1X(6)-C556*X(4)
91600              RES=RES/W5
91700              RETURN
91800    7         RES=X(6)-0.615
91900              RES=RES/W6
92000              RETURN
92100    205       IF (K.GT.8) GO TO 206
92200              RES=C7*X(2)*X(4)/(X(4)+(X(12)*(DUMMY4**TWOTH)/
92300             1DUMMY3))+(FFDZER-FZERO)/(THETR*DUMMY3)
92400    CCC       RES=RES/W7
92500              RETURN
92600    206       IF(K.GT.9)GO TO 207
92700              RES=C7*X(2)*X(4)/(X(4)+(X(12)*(DUMMY4**TWOTH)/
92800             1DUMMY3))+CT1*DASHI*N*X(6)*FZERO/(DUMMY3*(1-X(6)))+
92900             1(FFD(N)-F(N))*(DUMMY4**N)/(THETR*DUMMY3)
93000    CCC       RES=RES/FOLD(N)
93100              RETURN
93200    207       RES=C7*X(2)*X(4)/(X(4)+(X(12)*(DUMMY4**TWOTH)/
93300             1DUMMY3))+CT1*DASHI*N*X(6)*F(N-1)*(DUMMY4**(N-1))/(DUMMY3*
93400             1(1-X(6)))+(FFD(N)-F(N))*(DUMMY4**N)/(THETR*DUMMY3)
93500    CCC       RES=RES/FOLD(N)
93600              RETURN
93700              END
93800    C
93900    C
94000              FUNCTION CONFR(FF,YY,YY0)
94100    C*******************************************************************
94200    C
94300    C             THE INPUT FOR THIS SUBROUTINE ARE YY AND YY0.YY(I)
94400    C     REPRESENT THE   MOMENTS OF THE DISTRIBUTION(I=1,2,----) .YY0
94500    C     REPRESENTS ZEROTH MOMENT OF THE DISTRIBUTION .THE FUNCTION
94600    C     RETURNS THE FF TH MOMENT OF THE DISTRIBUTION.FOR DETAILS OF
94700    C     THE PROCEDURE EMPLOYED SEE CHAPTER 4.
94800    C
94900    C*******************************************************************
```

```
95000          DOUBLE PRECISION Y(6),YY(6),CCN(6),SSN(6),CALPHA(6),SALPHA(6)
95100          DOUBLE PRECISION YY0,AA,A,ALPHA,ALPHA1,ALKKA,ALPHN1,ALPHF1
95200          DOUBLE PRECISION ALPHFJ,ALPHJ1,FF,RATIO1
95300          NMAX=4
95400          AA=YY(2)*YY0/(YY(1)*YY(1))
95500          Y(1)=1.0/(AA-1.0)
95600          Y(2)=AA*Y(1)*Y(1)
95700          ALPHA=Y(1)-1.
95800          ALPHA1=Y(1)
95900          IF(ALPHA1.LT.0) GO TO 450
96000   C
96100   C
96200          DO 118 IK=3,NMAX
96300          KI=IK
96400          Y(KI) =((ALPHA1*YY0/YY(1))**KI)*YY(KI)/YY0
96500   118    CONTINUE
96600          ALPHF1=ALPHA+FF+1.0
96700   C
96800          DO 100 NMMM=3,NMAX
96900          N=NMMM
97000          CALPHA(N)=1
97100          DO 100 KKJJ=1,N
97200          KK=KKJJ
97300          NKK=N-KK
97400          ALKKA=ALPHA+KK+1.
97500          CCN(KK)=((-1)**KK)*FACTO(N)*GAMM(ALPHA1)/(FACTO(KK)*
97600         1FACTO(NKK)*GAMM(ALKKA))
97700          CALPHA(N)=CALPHA(N)+CCN(KK)* Y(KK)
97800   100    CONTINUE
97900          DO 201 NMMM=3,NMAX
98000          N=NMMM
98100          SALPHA(N)=0.0
98200          ALPHN1=N+ALPHA+1.
98300          SALPHA(N)=GAMM(ALPHN1)*GAMM(ALPHF1)/(FACTO(N)*GAMM(ALPHA1))
98400          DO 201 JJK=1,N
98500          JJ=JJK
98600          ALPHFJ=ALPHA+FF+JJ+1.0
98700          ALPHJ1=ALPHA+JJ+1.0
98800          ALPHN1=ALPHA+N+1.0
98900          NJJ=N-JJ

00100          SSN(JJ)=((-1.0)**JJ)*GAMM(ALPHN1)*GAMM(ALPHFJ)/(FACTO(NJJ)*
00200         1FACTO(JJ)*GAMM(ALPHJ1))
00300          SALPHA(N)=SALPHA(N)+SSN(JJ)
00400
00500
00600   201    CONTINUE
00700          A=GAMM(ALPHF1)
00800          DO 400 NMMM=3,NMAX
00900          N=NMMM
```

```
01000            A=A+CALPHA(N)*SALPHA(N)
01100     400    CONTINUE
01200            A=A/GAMM(ALPHA1)
01300            CONFR=((YY(1)/(YY0*ALPHA1))**FF)*A*YY0
01400            RETURN
01500     450    RATIO1=YY(1)/YY0
01600            CONFR=YY0*(RATIO1**FF)
01700            RETURN
01800            END
01900     C
02000     C
02100            FUNCTION FACTO(N)
02200     C**********************************************************************
02300     C
02400     C               THIS FUNCTION REPORTS THE VALUE OF FACTORIAL OF AN
02500     C        INTEGER VARIABLE N.
02600     C
02700     C**********************************************************************
02800            FACTO=1
02900            IF(N.EQ.0) GO TO 2
03000            DO 1 JK=1,N
03100            FACTO=FACTO*JK
03200     1      CONTINUE
03300     2      RETURN
03400            END
03500     C
03600     C
03700            FUNCTION GAMM(XX)
03800     C**********************************************************************
03900     C
04000     C               THIS FUNCTION REPORTS THE VALUE OF GAMMA FUNCTION OF
04100     C        THE VARIABLE XX.
04200     C
04300     C**********************************************************************
04400            DOUBLE PRECISION XX,X,Y
04500            IF(XX-57.0) 6,6,4
04600
04700     4      TER=2
04800            GAMM=1.E+30
04900            RETURN
05000     6      X=XX
05100            ERR=1.0E-06
05200            TER=0
05300            GAMM=1.0
05400            IF(X-2.0)50,50,15
05500     10     IF(X-2.0) 110,110,15
05600     15     X=X-1.
05700            GAMM=GAMM*X
05800            GO TO 10
05900     50     IF(X-1.0) 60,120,110
```

```
06000   C           SEE IF X IS NEAR NEGATIVE INTEGER OR ZERO
06100   60          IF(X-ERR) 62,62,80
06200   62          Y=FLOAT(INT(X))-X
06300               IF (ABS(Y)-ERR) 130,130,64
06400   64          IF(1.0-Y-ERR) 130,130,70
06500   C           X NOT NEAR A NEGATIVE INTEGER OR ZERO
06600   70          IF(X-1.0) 80,80,110
06700   80          GAMM=GAMM/X
06800               X=X+1.0
06900               GO TO 70
07000   110         Y=X-1.0
07100               GY=1.0+Y*(-0.5771017+Y*(+0.9858540+Y*(-0.8764218+Y*
07200              1(+0.8328212+Y*(-0.5684729+Y*(0.2548205+Y*(-.05149930)))))))
07300               GAMM=GAMM*GY
07400   120         RETURN
07500   130         IER=1
07600               RETURN
07700               END
07800   C
07900   C
08000               SUBROUTINE SET1 (N,FFD,ANUMBR,THETR,C22,C33,C78,C79,C44,T,B)
08100   C********************************************************************
08200   C
08300   C               THIS SUBROUTINE CALCULATES THE ELEMENTS OF COEFFICIENT
08400   C           MATRIX OF THE SET LINEAR EQUATIONS IN  SECOND SUBGROUP OF THE
08500   C           GROUP 1.COEFFICIENT MATRIX IS REPRESENTED BY T.B(I,1)IS THE
08600   C           VECTOR OF CONSTANTS .
08700   C
08800   C********************************************************************
08900               DOUBLE PRECISION T(25,25),B(25, 1),FFD(0:20)
09000               DOUBLE PRECISION ANUMBR,THETR,C22,C33,C78,C79,C44,THETVE,BRACE5
09100               N1=N-1
09200               THETVE=1./THETR
09300               DO 104 IROW=1,N
09400               DO 104 ICOL=1,N
09500               T(IROW,ICOL)=0.0
09600   104         CONTINUE
09700   C
09800               DO 105 IROW=1,N
09900               IF(IROW.EQ.N) GO TO 433
10000               ICOL=IROW
10100               ICOLM2=ICOL-2
10200               ICOLM1=ICOL-1
10300               ICOLP1=ICOL+1
10400               ICOLP2=ICOL+2
10500   C
10600               NEST=ICOLM2
10700               IF(NEST.LE.-1) GO TO 102
10800               IF(NEST.LE.0) GO TO 103
10900               T(IROW,NEST)=C22
```

```
11000   103     NEST=ICOLM1
11100           T(IROW,NEST)=C78
11200   102     NEST=ICOL
11300           T(IROW,NEST)=-C78-C33*(IROW-1)-C44*(IROW-1)*(IROW-2)-THETVE
11400          1-C22
11500           NEST=ICOLP1
11600           IF(NEST.GT.N) GO TO 105
11700           T(IROW,NEST)=C33*ICOL
11800           NEST=ICOLP2
11900           IF(NEST.GT.N) GO TO 105
12000           T(IROW,NEST)=ICOL*ICOLP1*C44
12100           GO TO 105
12200   433     DO 434 KI=1,N
12300           IK=KI
12400           T(IROW,IK)=1.0
12500   434     CONTINUE
12600   105     CONTINUE
12700   C
12800           BRACE5=THETR*ANUMBR
12900           DO 439 IROW=1,N
13000           IF(IROW.EQ.N) GO TO 438
13100           IF(IROW.EQ.2) GO TO 437
13200           B(IROW,1)=-FFD(IROW-1)/BRACE5
13300           GO TO 439
13400   437     B(IROW,1)=-FFD(IROW-1)/BRACE5-C79/ANUMBR
13500           GO TO 439
13600   438     B(IROW,1)=+1.0
13700   439     CONTINUE
13800           RETURN
13900           END
14000   C
14100   C
14200           SUBROUTINE LIN(NUMEQ,FFD,ANUMBR,THETR,C22,C33,C78,C79,C44,NX,
14300          1DASHI2,NTEMP3,B,IPRIN)
14400   C*********************************************************************
14500   C
14600   C               THIS SUBROUTINE SOLVES THE EQUATIONS OF SECOND SUBGROUP
14700   C       OF THE FIRST GROUP OF EQUATIONS.INITIALLY 'NUMEQ' EQUATIONS ARE
14800   C       SOLVED BY CALLING SET1 TO OBTAIN COEFFICIENT MATRIX AND MATIN
14900   C       TO THIS EQUATONS.THE ANSWER IS OBTAINED AS B(I,1) I=1,---NUMEQ.
15000   C       THE LAST TWO VALUES OF THE SET ARE TESTED TO BE LESS THAN
15100   C       1.0E-07  AND THE PROCESS IS REPEATED BY INCREASING THE NUMBER OF
15200   C       EQUATIONS BY 2.FOR DETAILS SEE CHAPTER 4.
15300   C
15400   C*********************************************************************
15500           DOUBLE PRECISION  T(25,25),B(25, 1),FFD(0:20)
15600           DOUBLE PRECISION ANUMBR,THETR,C22,C33,C78,C79,C44,DASHI2
15700           DOUBLE PRECISION SUM3,SUM4
15800           NTEMP=NUMEQ
15900   334     CALL SET1(NTEMP,FFD,ANUMBR,THETR,C22,C33,C78,C79,C44,T,B)
```

```
16000              M=1
16100              CALL MATIN(T,NTEMP,B,M,DETER)
16200              NTEMP1=NTEMP-1
16300 ·            NTEMP2=NTEMP-2
16400              IF(B(NTEMP1,1).GE.1.0E-07) GO TO 338
16500              IF(B(NTEMP2,1).GE.1.0E-07) GO TO 338
16600              NTEMP3=NTEMP-3
16700              IF(IPRIN.EQ.0) GO TO 339
16800              WRITE(NX,1119) (NTEMP3,(B(IK,1),IK=1,NTEMP3))
16900    1119      FORMAT(/5X,'ANSWER FROM LIN'//5X,'NTEMP3= ',I2//,2(5X,E15.8 ))
17000              GO TO 339
17100    338       NTEMP=NTEMP+2
17200              GO TO 334
17300    339       SUM3=0.0
17400              SUM4=0.0
17500              DO 335 IK=1,NTEMP3
17600              KI=IK
17700              SUM3=SUM3+B(KI,1)*(KI-1)
17800              SUM4=SUM4+B(KI,1)
17900    335       CONTINUE
18000              DASHI2=SUM3/SUM4
18100              RETURN
18200              END
18300 C
18400 C
18500              SUBROUTINE MATIN(A,N,B,M,DETER)
18600 C*******************************************************************
18700 C
18800 C                   THIS SUBROUTINE INVERTS THE COEFFICIENT MATRIX OF A SET
18900 C         OF LINEAR EQUATIONS.A REPRESENTS THE COEFFICIENT MATRIX .N IS
19000 C         THE NUMBER OF EQUATIONS.PUT M=1 IF ONLY SOLUTION OF EQUATIONS IS
19100 C         NEEDED.THE SOLUTION IS REPRESENTED BY B(I,1),I=1,----,N .
19200 C
19300 C*******************************************************************
19400              DOUBLE PRECISION B(25,1)
19500              DOUBLE PRECISION A(25,25)
19600              DOUBLE PRECISION AMAX,SWAP,PIVOT,DETRM,T
19700              DIMENSION IPIVT(40),INDEX(40,2)
19800              EQUIVALENCE(IROW,JROW),(ICOLM,JCOLM),(AMAX,T,SWAP)
19900    10        DETER=1.0
20000    15        DO 20 J=1,N

00100    20        IPIVT(J)=0
00200    30        DO 550 I=1,N
00300    40        AMAX=0.0
00400    45        DO 105 J=1,N
00500    50        IF(IPIVT(J)-1) 60,105,60
00600    60        DO 100 K=1,N
00700    70        IF(IPIVT(K)-1) 80,100,740
00800    80        IF(AMAX-ABS(A(J,K))) 85,100,100
```

```
00900      85        IROW=J
01000      90        ICOLM=K
01100     100        CONTINUE
01200     105        CONTINUE
01300     110        IPIVT(ICOLM)=IPIVT(ICOLM)+1
01400     130        IF(IROW-ICOLM) 140,260,140
01500     140        DETER=-DETER
01600     150        DO 200 L=1,N
01700     160        SWAP=A(IROW,L)
01800     170        A(IROW,L)=A(ICOLM,L)
01900     200        A(ICOLM,L)=SWAP
02000     205        IF(M) 260,260,210
02100     210        DO 250 L=1,M
02200     220        SWAP=B(IROW,L)
02300     230        B(IROW,L)=B(ICOLM,L)
02400     250        B(ICOLM,L)=SWAP
02500     260        INDEX(I,1)=IROW
02600     270        INDEX(I,2)=ICOLM
02700     310        PIVOT=A(ICOLM,ICOLM)
02800     320        DETRM=DETER*PIVOT
02900     330        A(ICOLM,ICOLM)=1.0
03000     340        DO 350 L=1,N
03100     350        A(ICOLM,L)=A(ICOLM,L)/PIVOT
03200     355        IF(M) 380,380,360
03300     360        DO 370 L=1,M
03400     370        B(ICOLM,L)=B(ICOLM,L)/PIVOT
03500     380        DO 550 L1=1,N
03600     390        IF(L1-ICOLM) 400,550,400
03700     400        T=A(L1,ICOLM)
03800     420        A(L1,ICOLM)=0.0
03900     430        DO 450 L=1,N
04000     450        A(L1,L)=A(L1,L)-A(ICOLM,L)*T
04100     455        IF(M) 550,550,460
04200     460        DO 500 L=1,M
04300     500        B(L1,L)=B(L1,L)-B(ICOLM,L)*T
04400     550        CONTINUE
04500     600        DO 710 I=1,N
04600     610        L=N+1-I
04700     620        IF(INDEX(L,1)-INDEX(L,2)) 630,710,630
04800     630        JROW=INDEX(L,1)
04900     640        JCOLM=INDEX(L,2)
05000     650        DO 705 K=1,N
05100     660        SWAP=A(K,JROW)
05200     670        A(K,JROW)=A(K,JCOLM)
05300     700        A(K,JCOLM)=SWAP
05400     705        CONTINUE
05500     710        CONTINUE
05600               DO 11 K=1,N
05700               IF(IPIVT(K)-1) 12,11,12
05800      11        CONTINUE
```

```
10900          SUM1=0.0
11000          DO 467 KI1=0,MOMENT
11100          KI=KI1
11200          IMONT=MOMENT-KI
11300          SUM1=SUM1+FACTO(MOMENT)*GSMAL(KI,IRAD)*GSMAL(IMONT,IRAD)/
11400         1(FACTO(IMONT)*FACTO(KI))
11500   467    CONTINUE
11600          TERM(IRAD)=C403*SUM1*XRAD(IRAD)/V
11700   468    CONTINUE
11800          DO 9 IROW=1,NTEMP3
11900          IF(IROW.GT.1) GO TO 7
12000          B(IROW,1)=-TERM(2)-C404*XLIVE(MOMENT,2)/V

00100         1-FDDED(MOMENT,0)*THETVE
00200          GO TO 9
00300   7      B(IROW,1)=-TERM(IROW+1)-C404*IROW*
00400         1XLIVE(MOMENT,IROW+1)/V-C63*TRP*XLIVE(MOMENT,IROW-1)
00500         1-C102*PHI*XLIVE(MOMENT,IROW-1)-FDDED(MOMENT,IROW-1)*THETVE
00600   9      CONTINUE
00700          RETURN
00800          END
00900   C
01000   C
01100          SUBROUTINE SET2(NTEMP3,C22,C33,C44,C78,C79,C63,C102,C101,TRP,
01200         1MOMENT,PHI,THETVE,FDLIV,XRAD,XLIVE,T,B)
01300   C**********************************************************************
01400   C
01500   C            THIS SUBROUTINE CALCULATES THE COEFFICIENT MATRIX OF THE
01600   C      SET OF SET LINEAR EQUATIONS  OF GROUP 2.(I.E. LIVE POLYMER CHAIN
01700   C      LENGTH DISTRIBUTION) COEFFICIENT MATRIX REPRESENTED T.VECTOR OF
01800   C      CONSTANTS IS  B(I,1) .
01900   C
02000   C**********************************************************************
02100          DOUBLE PRECISION T(25,25),B(25,1),XRAD(0:20)
02200          DOUBLE PRECISION FDLIV(5,20),XLIVE(0:5,20),TERM(20)
02300          DOUBLE PRECISION C22,C33,C44,C78,C63,C101,C102,PHI,TRP,THETVE
02400          DOUBLE PRECISION SUM1,C79
02500          NTEMP2=NTEMP3-1
02600          DO 4 IROW=1,NTEMP2
02700          DO 4 ICOL=1,NTEMP2
02800          T(IROW,ICOL)=0.0
02900   4      CONTINUE
03000   C
03100          DO 5 IROW=1,NTEMP2
03200          ICOL=IROW
03300          ICOLM2=ICOL-2
03400          ICOLM1=ICOL-1
03500          ICOLP1=ICOL+1
03600          ICOLP2=ICOL+2
03700   C
```

```
03800              NEST=ICOLM2
03900              IF(NEST.LE.-1) GO TO 2
04000              IF(NEST.LE.0)  GO TO 3
04100              T(IROW,NEST)=C22
04200      3       NEST=ICOLM1
04300             .T(IROW,NEST)=C78
04400      2       NEST=ICOL
04500              T(IROW,NEST)=-C78-C22-C33*ICOL-C63*TRP-C102*PHI-C44*ICOL*ICOLM1
04600             1-THETVE
04700              NEST=ICOLP1
04800              IF(NEST.GT.NTEMP2) GO TO 5
04900              T(IROW,NEST)=C33*ICOL
05000              NEST=ICOLP2
05100              IF(NEST.GT.NTEMP2) GO TO 5
05200              T(IROW,NEST)= C44*ICOLP1*ICOL
05300      5       CONTINUE
05400              DO 468 IRAD1=1,NTEMP2
05500              IRAD=IRAD1
05600              SUM1=0.0
05700              DO 467 KI1=1,MOMENT
05800              KI=KI1
05900              IMONT=MOMENT-KI
06000              SUM1=SUM1+FACTO(MOMENT)*XLIVE(IMONT,IRAD)/(FACTO(KI)*
06100             1FACTO(IMONT ))
06200      467     CONTINUE
06300              TERM(IRAD)=C101*PHI*SUM1
06400      468     CONTINUE
06500      C
06600              DO 9 IROW=1,NTEMP2
06700              IF(IROW.GT.1) GO TO 7
06800              B(IROW,1) =-C78*XRAD(0)-C102*PHI*1.0*XRAD(1)-THETVE*
06900             1FDLIV(MOMENT,1)-TERM(1)-C63*TRP*1.0*XRAD(1)-C79
07000              GO TO 9
07100      7       B(IROW,1)=-C78*XRAD(IROW-1)-2.0*C22*XRAD(IROW-2)-C102*PHI*
07200             1IROW*XRAD(IROW)-THETVE*FDLIV(MOMENT,IROW)-TERM(IROW)
07300             1-C63*TRP*IROW*XRAD(IROW)
07400      9       CONTINUE
07500              RETURN
07600              END
07700      C
07800      C
07900              FUNCTION DIFF2(AMV,V)
08000      C***************************************************************************
08100      C
08200      C          THE INPUT FOR THIS FUNCTION CONSISTS OF AMV AND V,AMV(I)
08300      C   ARE THE MOMENTS OF A DISTRIBUTION,(I=0,1,2----) THE FUNCTION
08400      C   RETURNS THE VALUE OF DISTRIBUTION FUNCTION EVALUATED AT V,
08500      C          FOR DETAILS SEE CHAPTER 4.
08600      C
08700      C***************************************************************************
```

```
08800          DOUBLE PRECISION AMV(0:5),GX(6),CALPHA(6),ALPHAL(6)
08900   C       AMV ARE FIRST FOUR MOMENTS OF DISTRIBUTION
09000   C       AMVZER IS THE ZEROTH ABSOLUTE MOMENT OF DISTRIBUTION
09100   C       X IS THE X CO-ORDINATE OF WHICH
09200   C       THE FUNCTION IS TO BE EVALUATED.
09300   C       RESULT IS THE FUNCTION VALUE CALCULATED AT
09400   C       AFORESAID X CO-ORDINATE @V@.
09500   C
09600   C       INPUT PORTION-----
09700   C       CONVERT MV INTO GX(I,E, MOMENTS OF NORMALIEED
09800   C       DISTRIBUTION.)
09900   C
10000          DOUBLE PRECISION AMVZER,PDI,ALPHA,ALPHA1,ALPIT1,ALPIK1,XMINU,SUM
10100          DOUBLE PRECISION  V
10200          AMVZER=AMV(0)
10300         PDI=AMV(2)*AMVZER/(AMV(1)*AMV(1))
10400         GX(1)=1.0/(PDI-1.0)
10500         GX(2)=PDI*GX(1)*GX(1)
10600         ALPHA=GX(1)-1.0
10700         ALPHA1=GX(1)
10800         NMAX=4
10900         DO 118 KI=3,NMAX
11000         IK=KI
11100         GX(IK)=((ALPHA1*AMVZER/AMV(1))**IK)*AMV(IK)/AMVZER
11200   118   CONTINUE
11300   C       CONVERT ABSOLUTE CO-ORDINATE V TO THE CORRESSPONDING
11400   C       NORMALISED CO-ORDINATE X.
11500         X=(ALPHA1*AMVZER/AMV(1))*V
11600   C       CALCULATE CALPHA.
11700         DO 100 KI=3,NMAX
11800         IK=KI
11900         CALPHA(IK)=1.0
12000         DO 100 ITAM=1,IK
12100         ITEMP=ITAM
12200         IKITAM=IK-ITEMP
12300         ALPIT1=ALPHA1+ITEMP
12400           CALPHA(IK)=CALPHA(IK)+((-1)**ITEMP)*FACTO(IK)*GAMM(ALPHA1)*
12500           1GX(ITEMP)/(FACTO(IKITAM)*FACTO(ITEMP)*GAMM(ALPIT1))
12600   100   CONTINUE
12700   C       CALCULATION OF MODIFIED LAGURE@ POLYNOMIAL
12800         DO 200 KI=3,NMAX
12900         IK=KI
13000         ALPIK1=ALPHA1+IK
13100         ALPHAL(IK)=GAMM(ALPIK1)/(GAMM(ALPHA1)*FACTO(IK))
13200         DO 200 ITAM=1,IK
13300         ITEMP=ITAM
13400         IKITAM=IK-ITEMP
13500         ALPIT1=ALPHA1+ITEMP
13600         ALPHAL(IK)=ALPHAL(IK)+((-1)**ITEMP)*GAMM(ALPIK1)*
13700         1(X**ITEMP)/(FACTO(IKITAM)*GAMM(ALPIT1)*FACTO(ITEMP))
```

```
13800   200     CONTINUE
13900   C          CALCULATION OF NORMALISED DISTRIBUTION FUNCTON.
14000           SUM=1.0
14100           DO 300 KI=3,NMAX
14200           IK=KI
14300           SUM=SUM+CALPHA(IK)*ALPHAL(IK)
14400   300     CONTINUE
14500           XMINU=-X
14600           GNORM=SUM*EXP(XMINU)*(X**ALPHA)/GAMM(ALPHA1)
14700   C          CALCULATION OF VALUE OF TJE ABSOLUTE FUNX.
14800           DIFF2=GNORM*ALPHA1*AMVZER*AMVZER/AMV(1)
14900           RETURN
15000           END
15100   C
15200   C
15300   C
15400   C
15500           SUBROUTINE CENTR(N,M,K,IEV1,I,XC,X,K1)
15600   C************************************************************************
15700   C
15800   C                   THIS SUBROUTINE IS REQUIRED FOR OPTIMIZATION,IT
15900   C          CALCULATES THE COEFFICIENTS OF THE CENTROID OF THE COMPLEX,
16000   C
16100   C************************************************************************
16200           DOUBLE PRECISION X(K,M),XC(N)
16300           DO 20 J=1,N
16400           XC(J)=0.0
16500           DO 10 IL=1,K1
16600           XC(J)=XC(J)+X(IL,J)
16700   10      CONTINUE
16800           RK=K1
16900           XC(J)=(XC(J)-X(IEV1,J))/(RK-1.0)
17000   20      CONTINUE
17100           RETURN
17200           END
17300   C
17400   C
17500           SUBROUTINE CHECK(N,M,K,X,G,H,I,KODE,XC,DELTA,K1)
17600   C************************************************************************
17700   C
17800   C                   THIS SUBROUTINE IS REQUIRED FOR OPTIMIZATION,IT CHECKS
17900   C          IF A GIVEN POINT SATISFIES THE GIVEN SET OF CONSTRAINTS AND
18000   C          AND CALCULATES A FEASIBLE  VERTEX FOR THE COMPLEX.
18100   C
18200   C************************************************************************
18300           DOUBLE PRECISION X(K,M),G(M),H(M),XC(N)
18400   10      KT=0
18500           CALL CONST(N,M,K,X,G,H,I)
18600           DO 50 J=1,N
18700           IF(X(I,J)-G(J)) 20,20,30
```

```
18800    20      X(I,J)=G(J)+DELTA
18900            GO TO 50
19000    30      IF(H(J)-X(I,J)) 40,40,50
19100    40      X(I,J)=H(J)-DELTA
19200    50      CONTINUE
19300            NO=5
19400    C       WRITE (NO,*) ((J,X(1,J),XC(J),X(2,J)) ,J=1,N)
19500            IF(KODE) 110,110,60
19600    60      NN=N+1
19700            DO 100 J=NN,M
19800            CALL CONST(N,M,K,X,G,H,I)
19900            IF(X(I,J)-G(J)) 80,70,70
20000    70      IF(H(J)-X(I,J)) 80,100,100
20100    80      IEV1=I
20200            KT=1
20300            CALL CENTR(N,M,K,IEV1,I,XC,X,K1)
20400            DO 90 JJ=1,N
20500            X(I,JJ)=(X(I,JJ)+XC(JJ))/2.0
20600    90      CONTINUE
20700    100     CONTINUE
20800            IF(KT) 110,110,10
20900    110     RETURN
21000            END
21100    C
21200    C
21300            SUBROUTINE CONSX(N,M,K,ITMAX,ALPHA,BETA,GAMMA,DELTA,X,R,F,
21400           1IT,IEV2,NO,G,H,XC,IPRIN,DASHI,ONETH,TWOTH,BRACE1,BRACE2,
21500           1BRACE3,C50,C51,C52,C53,C54,C55,C556,C56,CT1,C7,THETR,DUMMY,XFD,
21600           1C65,SAI)
21700    C*************************************************************************
21800    C
21900    C               THIS SUBROUTINE IS USED TO FIND MAXIMA/MINIMA OF A
22000    C       CONSTRAINED MULTIVARIABLE FUNCTION BY BOX'S METHOD,IT IS USED
22100    C       TO FIND INITIAL GUESSES TO SOLVE THE EQUATIONS OF FIRST SUBGROUP
22200    C       OF GROUP 1,FOR DETAILS SEE CHAPTER 4 AND REF 32
22300    C
22400    C       DISCRIPTION OF PRINCIPAL VARIABLES____
22500    C       ALPHA_____COMPLEX EXPANSION PARAMETER,
22600    C                                          (USUALLY 1.3)
22700    C       BETA_____CONVERGENCE CRITERION
22800    C       GAMMA_____CONVERGENCE CRITERION
22900    C       DELTA_____USED TO FIND A FEASIBLE POINT
23000    C                                          WHEN THE CALCULATED VERTEX OF TH
23100    C                                          COMPLEX IS FOUND TO BE OUTSIDE
23200    C                                          ANY CONSTRAINT.
23300    C       N_____NUMBER OF VARIABLES,
23400    C       M_____NUMBER OF CONSTRAINTS,
23500    C       K_____NUMBER OF VERTICES,
23600    C       ITMAX_____MAXIMUM NUMBER OF ITERATIONS
23700    C                                          ALLOWED FOR OPTIMIZATION,
```

```
23800    C          R------------------------------RANDOM NUMBERS.
23900    C          F------------------------------OBJECTIVE FUNCTION.
24000    C          X(I)---------------------------VECTOR OF  VARIABLES
24100    C
24200    C*****************************************************************
24300             DOUBLE PRECISION ONETH,TWOTH,BRACE1,BRACE3,BRACE2,THETR,DUMMY
24400             DOUBLE PRECISION C50,C51,C52,C53,C54,C55,C556,C56,C7,CT1,C65
24500             DOUBLE PRECISION DASHI,DASHI1,DASHI2,SAI
24600             INTEGER GAMMA
24700             DOUBLE PRECISION X(K,M),F(K),G(M),H(M),XC(N),XFD(N)
24800             DIMENSION R(K,N)
24900             IT=1
25000             IPRIN1=IPRIN
25100             KODE=0
25200             IF(M-N)20,20,10
25300       10    KODE=1
25400       20    CONTINUE
25500             DO 40 II=2,K
25600             DO 30 J=1,N
25700             X(II,J)=0.0
25800       30    CONTINUE
25900       40    CONTINUE
26000             DO 65 II=2,K
26100             DO 50 J=1,N
26200             I=II
26300             CALL CONST(N,M,K,X,G,H,I)
26400             X(II,J)=G(J)+R(II,J)*(H(J)-G(J))
26500       50    CONTINUE
26600             K1=II
26700             CALL CHECK(N,M,K,X,G,H,I,KODE,XC,DELTA,K1)
26800             IF(II-2)51,51,55
26900       51    IF(IPRIN)52,65,52
27000       52    WRITE(NO,018)
27100       18    FORMAT(1H1//2X,'COORDINATES OF INITIAL COMPLEX')
27200             IO=1
27300             WRITE (NO,159) (IO,J,X(IO,J),J=1,N)
27400      159    FORMAT(/3(2X,2HX(,I2,1H,,I2,4H) = ,E13.6))
27500       55    IF(IPRIN) 56,65,56
27600       56    WRITE (NO,159) (II,J,X(II,J),J=1,N)
27700       65    CONTINUE
27800             K1=K
27900             DO 70 IL=1,K
28000             I=IL
28100             CALL FUNC(N,M,K,X,F,I,DASHI,BRACE1,BRACE2,
28200            1BRACE3,C50,C51,C52,C53,C54,C55,C556,C56,CT1,C7,THETR,DUMMY,XFD,
28300            1C65,SAI)
28400       70    CONTINUE
28500             KOUNT=1
28600             IA=0
28700             IF(IPRIN) 72,80,72
```

```
28800    72          WRITE(NO,21)
28900    21          FORMAT(/2X,'VALUES OF THE FUNCTION')
29000                WRITE(NO,22) (J,F(J),J=1,K)
29100    22          FORMAT(/,3(2X,2HF(,I2,4H) = ,E13.6))
29200    80          IEV1=1
29300                DO 100 ICM=2,K
29400                IF(F(IEV1)-F(ICM)) 100,100,90
29500    90          IEV1=ICM
29600    100         CONTINUE
29700                IEV2=1
29800                DO 120 ICM=2,K
29900                IF(F(IEV2)-F(ICM)) 110,110,120
30000    110         IEV2=ICM
30100    120         CONTINUE
30200                IF(F(IEV2)-(F(IEV1)+BETA)) 140,130,130
30300    130         KOUNT=1
30400                GO TO 150
30500    140         KOUNT=KOUNT+1
30600                IF(KOUNT-GAMMA) 150,240,240
30700    150         CALL CENTR(N,M,K,IEV1,I,XC,X,K1)
30800                DO 160 JJ=1,N
30900                X(IEV1,JJ)=(1.0+ALPHA)*XC(JJ)-ALPHA*X(IEV1,JJ)
31000    160         CONTINUE
31100                I=IEV1
31200                CALL CHECK(N,M,K,X,G,H,I,KODE,XC,DELTA,K1)
31300                CALL FUNC(N,M,K,X,F,I,DASHI,BRACE1,BRACE2,
31400                1BRACE3,C50,C51,C52,C53,C54,C55,C556,C56,CT1,C7,THETR,DUMMY,XFD,
31500                1C65,SAI)
31600    170         IEV2=1
31700                DO 190 ICM=2,K
31800                IF(F(IEV2)-F(ICM)) 190,190,180
31900    180         IEV2=ICM
32000    190         CONTINUE
32100                IF(IEV2-IEV1) 220,200,220
32200    200         DO 210 JJ=1,N
32300                X(IEV1,JJ)=(X(IEV1,JJ)+XC(JJ))/2.0
32400    210         CONTINUE
32500                I=IEV1
32600                CALL CHECK(N,M,K,X,G,H,I,KODE,XC,DELTA,K1)
32700                CALL FUNC(N,M,K,X,F,I,DASHI,BRACE1,BRACE2,
32800                1BRACE3,C50,C51,C52,C53,C54,C55,C556,C56,CT1,C7,THETR,DUMMY,XFD,
32900                1C65,SAI)
33000                GO TO 170
33100    220         CONTINUE
33200                IF(IT/50*50.NE.IT) IPRIN=0
33300                MTYPE=1
33400                IF(IT/50*50.NE.IT) MTYPE=0
33500                IF(MTYPE.EQ.0) GO TO 4511
33600    C           IF(IT.GT.300) IPRIN=1
33700                TYPE *,IT
```

```
33800    4511      IF(IPRIN) 230,228,230
33900    230       WRITE(NO,23)IT
34000    23        FORMAT(//,2X,'ITERATION NUMBER',I5)
34100              WRITE(NO,24)
34200    24        FORMAT(/,2X,'CO-ORDINATES OF CORRECTED POINT')
34300              WRITE(NO,159) (IEV1,JC,X(IEV1,JC),JC=1,N)
34400              WRITE(NO,21)
34500              WRITE(NO,22)(I,F(I),I=1,K)
34600              WRITE(NO,25)
34700    25        FORMAT(/,2X,'CO-ORDINATES OF THE CENTROID')
34800              WRITE(NO,26)(JC,XC(JC),JC=1,N)
34900    26        FORMAT(/,3(2X,2HX(,I2,6H,C) = ,E13,6))
35000    228       IT=IT+1
35100              IPRIN=IPRIN1
35200              IF(IT-ITMAX) 80,80,240
35300    240       RETURN
35400              END
35500    C
35600    C
35700              SUBROUTINE CONST(N,M,K,X,G,H,I)
35800              DOUBLE PRECISION X(14,16),G(16),H(16)
35900    C********************************************************************
36000    C
36100    C              THIS SUBROUTINE DEFINES VARIOUS EXPLICIT AND IMPLICIT
36200    C        CONSTRAINTS.
36300    C        G(I)_____LOWER CONSTRAINTS.
36400    C        H(I)_____UPPER CONSTRAINTS.
36500    C        I=1,----N_____EXPLICIT CONSTRAINTS.
36600    C        I=N+1,----M_____IMPLICIT CONSTRAINTS.
36700    C
36800    C********************************************************************
36900              G(1)=0,0
37000              H(1)=4,9
37100              G(2)=0,0
37200              H(2)=1,0
37300              G(3)=0,0
37400              H(3)=1,92
37500
37600                  G(4)=1,0E-06
37700              H(4)=1,0E+06
37800              G(5)=1,0E-06
37900              H(5)=1,0E+06
38000              G(6)=,01
38100              H(6)=1,0
38200              G(7)=1,0E-06
38300              H(7)=1,0E+03
38400              G(8)=1,0E-06
38500              H(8)=1,0E+04
38600              G(9)=1,0E-06
38700              H(9)=1,0E+05
```

```
38800          G(10)=1.0E-06
38900          H(10)=1.0E+06
39000          G(11)=1.0E-07
39100          H(11)=1.0E+07
39200          IF(N-M) 1,2,1
39300    1     X(I,12)=X(I,7)*X(I,9)/X(I,8)**2
39400          G(12)=1.0
39500          H(12)=10.0
39600          X(I,13)=X(I,8)/X(I,7)
39700          G(13)=0.5
39800          H(13)=10.0
39900          X(I,14)=X(I,9)/X(I,8)
40000          G(14)=1.0
40100          H(14)=10
40200          X(I,15)=X(I,10)/X(I,9)
40300          G(15)=1.0
40400          H(15)=10.0
40500          X(I,16)=X(I,11)/X(I,10)
40600          G(16)=1.0
40700          H(16)=10.0
40800    2     RETURN
40900          END
41000  C
41100  C
41200          SUBROUTINE FUNC (INN,M,K,XDUMY,F,J,DASHI,BRACE1,BRACE2,
41300         1BRACE3,C50,C51,C52,C53,C54,C55,C556,C56,CT1,C7,THETR,DUMMY,XFD)
41400  C*********************************************************************
41500  C
41600  C              THIS SUBROUTINE RETURNS THE VALUE OF THE OBJECTIVE
41700  C       FUNCTION TO BE OPTIMIZED.THE OBJECTIVE FUNCTION IS MADE FROM
41800  C       R.H.S. OF EQUATIONS OF FIRST SUBGROUP OF GROUP 1.
41900  C
42000  C*********************************************************************
42100          DOUBLE PRECISION XDUMY(K,M),F(K),XFD(11),Z(6),FFD(6),FOLD(6)
42200          DOUBLE PRECISION X(25),RUMY(6)
42300          DOUBLE PRECISION ONETH,TWOTH,BRACE1,BRACE3,BRACE2,THETR,DUMMY
42400          DOUBLE PRECISION C50,C51,C52,C53,C54,C55,C556,C56,C7,CT1
42500          DOUBLE PRECISION DASHI,RES,FZERO,FFDZER,FIRST,SECOND,THIRD
42600          DOUBLE PRECISION FOURTH,FIFTH,SIX,SEVENT,SUM
42700          NMAX=4
42800  C
42900  C
43000          ONETH=1.0/3.
43100          TWOTH=2.0/3.
43200  C
43300          W1=8.08E-04
43400          W2=47.69
43500          W3=3.807E-02
43600          W4=2.937E+06
43700          W5=7.0998E-01
```

```
43800              W6=0.63
43900              W7=0.1981
44000              FOLD(1)=8.5111E+02
44100              FOLD(2)=7.3E+06
44200              FOLD(3)=9.432E+10
44300              FOLD(4)=1.622E+15
44400     C
44500              DUMMY1=1.0E+06
44600              DUMMY2=1.0E-02
44700              DUMMY3=1.0E-03
44800              DUMMY4=4.2985E+03
44900     C
45000     C        X ARE THE OUTPUT VARIABLES FROM THE REACTOR
45100     C        XFD ARE THE FEED CHARECTARISTIC OF AFORESAID VARIABLES
45200              DO 450 IK=1,INN
45300              KI=IK
45400              X(KI)=XDUMY(J,KI)
45500     450      CONTINUE
45600     C
45700              N=K-7
45800              FZERO=X(7)
45900              FFDZER=XFD(7)
46000              DO 210 INTEMP=1,NMAX
46100              NTEM=INTEMP
46200              Z(NTEM)=X(NTEM+7)
46300              FFD(NTEM)=XFD(NTEM+7)
46400     210      CONTINUE
46500     1        RES=(XFD(1)-X(1))/THETR
46600              FIRST=RES/W1
46700     2        RES=C56*(XFD(2)-X(2))/THETR-X(2)
46800              SECOND=RES/W2
46900     3        RES=(XFD(3)-X(3))/(THETR*DUMMY2)-C53*X(6)*(FZERO/DUMMY3)*DASHI
47000              THIRD=RES/W3
47100     4        RES=C50*X(1)-X(4)-C51-CONFR(TWOTH,Z,FZERO)*(DUMMY4**TWOTH)/
47200             1DUMMY3-C52*X(5)/DUMMY1
47300              FOURTH=RES/W4
47400     5        RES=C54*X(3)/DUMMY2-(X(5)/DUMMY1)-C55*(Z(1)*DUMMY4/DUMMY3)*
47500             1X(6)-C556*X(4)
47600              FIFTH=RES/W5
47700     6        RES=X(6)-0.615
47800              SIX=RES/W6
47900              RES=C7*X(2)*X(4)/(X(4)+(CONFR(TWOTH,Z,FZERO)*(DUMMY4**TWOTH)/
48000             1DUMMY3))+(FFDZER-FZERO)/(THETR*DUMMY3)
48100     C
48200     C********************************************************************
48300     C
48400     C        INSERT THIS LINE INSTEAD OF  PREVIOUS LINE FOR SECOND APPROACH
48500     C6       RES=1.0-X(6)+DLOG(X(6))+SAI*(1.0-X(6))+C65*X(7)/
48600     C       1(CONFR(ONETH,Z,FZERO)*(DUMMY4*ONETH))
48700     C
```

```
48800    C*************************************************************************
48900    C
49000            SEVENT=RES/W7
49100            DO 650 N=1,NMAX
49200            IF(N.GT.1) GO TO 207
49300            RES=C7*X(2)*X(4)/(X(4)+(CONFR(TWOTH,Z,FZERO)*(DUMMY4**TWOTH)/
49400           1DUMMY3))+CT1*DASHI*N*X(6)*FZERO/(DUMMY3*(1-X(6)))+
49500           1(FFD(N)-Z(N))*(DUMMY4**N)/(THETR*DUMMY3)
49600            RUMY(N)=RES/FOLD(N)
49700            GO TO 650
49800    207     RES=C7*X(2)*X(4)/(X(4)+(CONFR(TWOTH,Z,FZERO)*(DUMMY4**TWOTH)/
49900           1DUMMY3))+CT1*DASHI*N*X(6)*Z(N-1)*(DUMMY4**(N-1))/(DUMMY3*
50000           1(1-X(6)))+(FFD(N)-Z(N))*(DUMMY4**N)/(THETR*DUMMY3)
50100            RUMY(N)=RES/FOLD(N)
50200    650     CONTINUE
50300            SUM=0.0
50400            DO 550 IK=1,NMAX
50500            KI=IK
50600            SUM=SUM+ABS(RUMY(KI))
50700    550     CONTINUE
50800            F(J)=SUM+ABS(FIRST)+ABS(SECOND)+ABS(THIRD)+ABS(FOURTH)+
50900           1ABS(FIFTH)+ABS(SIX)+ABS(SEVENT)
51000            F(J)=-F(J)
51100            RETURN
51200            END
51300    C
51400    C
51500            SUBROUTINE QG10(XL,XU,Y,AMV)
51600    C*************************************************************************
51700    C
51800    C                THIS SUBROUTINE RETURNS THE VALUE OF INTEGRAL OF A
51900    C       FUNCTION EVALUATED IN THE RANGE XL TO XU.IT IS USED FOR
52000    C       CALCULATING THE CUMULATIVE DISTRIBUTION FROM A DIFFERENTIAL
52100    C       DISTRIBUTION.
52200    C
52300    C*************************************************************************
52400            DOUBLE PRECISION Y,XL,XU
52500            DOUBLE PRECISION AMV(0:5)
52600            DOUBLE PRECISION A,B,C
52700             A=0.5*(XU-XL)+XL
52800            B=(XU-XL)
52900            C=0.4869533*B
53000            Y=0.03333567*(DIFF2(AMV,A+C)+DIFF2(AMV,A-C))
53100            C=0.4325317*B
53200            Y=Y+0.07472567*(DIFF2(AMV,A+C)+DIFF2(AMV,A-C))
53300            C=0.3397048*B
53400            Y=Y+0.1095432*(DIFF2(AMV,A+C)+DIFF2(AMV,A-C))
53500            C=0.2166977*B
53600            Y=Y+0.1346334*(DIFF2(AMV,A+C)+DIFF2(AMV,A-C))
53700            C=0.07443717*B
```

```
53800          Y=B*(Y+0.1477621*(DIFF2(AMV,A+C)+DIFF2(AMV,A-C)))
53900          RETURN
54000          END
54100  C
54200  C
54300          SUBROUTINE DISTR(XL1,PART1,AMV,IWHICH,XCORD,YCORD,ZCORD,XCUM)
54400  C***********************************************************************
54500  C
54600  C              THIS SUBROUTINE GENERATES THE DISTRIBUTION FUNCTION FROM
54700  C       ITS MOMENT.INITIALLY THE FUNCTION IS EVALUATED AT STEPS OF
54800  C       PART1 STARTING WITH THE LOWER LIMIT OF XL1.THE USEFUL RANGE OF
54900  C       DISTRIBUTION IS THEN CALCULATED .THE USEFUL RANGE IS THEN
55000  C       DIVIDED INTO 50 INTERAVALS AND THE DISTRIBUTION FUNCTION IS
55100  C       CALCULATED AT THESE 51 POINTS.XCORD(I)IS THE SET OF X
55200  C       CO-ORDINATES AT WHICH THE FUNCTION CALCULATED,YCORD(I) IS THE
55300  C       SET OF FUNCTION VALUES AT THESE POINTS ON NUMBER BASIS.
55400  C       ZCORD(I) IS THE SET OF FUNCTION VALUES ON WT.BASIS,XCUM
55500  C       REPRESENTS THE SET OF NORMALISED CUMULATIVE DISTRIBUTION
55600  C       FUNCTION CALCULATED AT THE AFORESAID POINTS
55700  C       IWHICH=0----NORMALISED CUMULATIVE DISTRIBUTION IS NOT CALCULATED
55800  C
55900  C
56000  C***********************************************************************
56100          DOUBLE PRECISION XCORD(0:50),YCORD(0:50),ZCORD(0:50),XCUM(0:50)
56200          DOUBLE PRECISION STAR(50),AMV(0:5)
56300          DOUBLE PRECISION XL,XL1,XU1,XU,PART1,PART2,Y,YMAX
56400          DOUBLE PRECISION  RATIO,AREA,TOTAL
56500          XL=XL1
56600  C       TO FIND USEFUL RANGE OF DISTRIBUTION.
56700          YMAX=0.0
56800  1       Y=DIFF2(AMV,XL)
56900          IF(Y.GT.YMAX) GO TO 15
57000          RATIO=Y/YMAX
57100          IF(RATIO.GT.0.0001)GO TO 16
57200          GO TO 2
57300  15      YMAX=Y
57400  16      XL=XL+PART1
57500          GO TO 1
57600  C       UPPER USEFUL LIMIT FOR THE DISTRIBUTION HAS BEEN OBTAINED
57700  2       XU1=XL
57800          PART2=(XU1-XL1)/50.0
57900          XL=XL1
58000  C       TO FIND DIFFERENTIAL AND CUMULATIVE DISTRIBUTION
58100          XCORD(0)=XL1
58200          YCORD(0)=DIFF2(AMV,XL1)
58300          ZCORD(0)=XCORD(0)*YCORD(0)
58400          XCUM(0)=0.0
58500          AREA=0.0
58600          DO 3 KI=1,50
58700          XU=XL+PART2
```

```
58800          XCORD(KI)=XU
58900          YCORD(KI)=DIFF2(AMV,XU)
59000          ZCORD(KI)=XCORD(KI)*YCORD(KI)
59100          IF(IWHICH.EQ.0) GO TO 6
59200          CALL QG10(XL,XU,Y,AMV)
59300          STAR(KI)=Y
59400          AREA=AREA+STAR(KI)
59500     6    XL=XU
59600     3    CONTINUE
59700          IF(IWHICH.EQ.0) GO TO 5
59800     C    TO CALCULATE NORMALISED CUMULATIVE DISTRIBUTION
59900          TOTAL=AREA
60000          AREA=0.0
60100          DO 4 KI=1,50
60200          AREA=AREA+STAR(KI)
60300          XCUM(KI)=AREA/TOTAL
60400     4    CONTINUE
60500     5    RETURN
60600          END
60700     C
60800     C
60900          SUBROUTINE FUNXX(X,RES,K,DASHI,
61000         1ONETH,TWOTH,BRACE1,BRACE2,BRACE3,C50,C51,C52,C53,C54,C55,C556,
61100         1C56,C7,CT1,C65, DUMMY,THETR,SAI,XFD,X66)
61200     C*********************************************************************
61300     C              THIS SUBROUTINE IS SIMILAR TO FUNX,FOR SECOND APPROACH
61400     C    CHANGE THE NAME OF SUBROUTINE TO FUNX.
61500     C*********************************************************************
61600          DOUBLE PRECISION X(13),XFD(13),F(6),FFD(6),FOLD(6)
61700          DOUBLE PRECISION RES,DASHI,ONETH,TWOTH
61800          DOUBLE PRECISION BRACE1,BRACE2,BRACE3,DUMMY,THETR,SAI
61900          DOUBLE PRECISION C50,C51,C52,C53,C54,C55,C556,C56,C7,CT1,C65
62000          DOUBLE PRECISION FZERO,FFDZER,X66
62100     C    COMMON ONETH,TWOTH,BRACE1,BRACE2,BRACE3,C50,C51,C52,C53,C54,C55,
62200     C   1C556,C56,CT1,C7,THETR,DUMMY
62300          NMAX=4
62400     C
62500     C
62600     C
62700          W1=8.08E-04
62800          W2=47.69
62900          W3=3.807E-02
63000          W4=2.937E+06
63100          W5=7.0998E-01
63200          W6=0.63
63300          W7=0.1981
63400          FOLD(1)=8.5111E+02
63500          FOLD(2)=7.3E+06
63600          FOLD(3)=9.432E+10
63700          FOLD(4)=1.622E+15
```

```
63800    C
63900             DUMMY1=1.0E+06
64000             DUMMY2=1.0E-02
64100             DUMMY3=1.0E-03
64200             DUMMY4=4.2985E+03
64300    C
64400    C        X ARE THE OUTPUT VARIABLES FROM THE REACTOR
64500    C        XFD ARE THE FEED CHARECTARISTIC OF AFORESAID VARIABLES
64600             N=K-7
64700             FZERO=X(6)
64800             FFDZER=XFD(6)
64900             DO 210 INTEMP=1,NMAX
65000             NTEM=INTEMP
65100             F(NTEM)=X(NTEM+6)
65200             FFD(NTEM)=XFD(NTEM+6)
65300    210      CONTINUE
65400             IF(K.GT.6) GO TO 205
65500             GO TO (1,2,3,4,5,6),K
65600    1        CONH=CONFR(TWOTH,F,FZERO)
65700             RES=X(11)-CONH
65800             RETURN
65900    2        RES=(XFD(1)-X(1))/THETR
66000             RES=RES/W1
66100             RETURN
66200    3        RES=C56*(XFD(2)-X(2))/THETR-X(2)
66300             RES=RES/W2
66400             RETURN
66500    4        RES=(XFD(3)-X(3))/(THETR*DUMMY2)-C53*X66*(FZERO/DUMMY3)*DASHI
66600             RES=RES/W3
66700             RETURN
66800    5        RES=C50*X(1)-X(4)-C51-X(11)*(DUMMY4**TWOTH)/
66900            1DUMMY3-C52*X(5)
67000             RES=RES/W4
67100             RETURN
67200    6        RES=C54*X(3)/DUMMY2-(X(5))-C55*(F(1)*DUMMY4/DUMMY3)*
67300            1X66-C556*X(4)
67400             RES=RES/W5
67500             RETURN
67600    205      IF (K.GT.7) GO TO 206
67700             RES=C7*X(2)*X(4)/(X(4)+(X(11)*(DUMMY4**TWOTH)/
67800            1DUMMY3))+(FFDZER-FZERO)/(THETR*DUMMY3)
67900    CCC      RES=RES/W7
68000             RETURN
68100    206      IF(K.GT.8)GO TO 207
68200             RES=C7*X(2)*X(4)/(X(4)+(X(11)*(DUMMY4**TWOTH)/
68300            1DUMMY3))+CT1*DASHI*N*X66*FZERO/(DUMMY3*(1-X66))+
68400            1(FFD(N)-F(N))*(DUMMY4**N)/(THETR*DUMMY3)
68500    CCC      RES=RES/FOLD(N)
68600             RETURN
68700    207      RES=C7*X(2)*X(4)/(X(4)+(X(11)*(DUMMY4**TWOTH)/
```

```
68800              1DUMMY3))+CT1*DASHI*N*X66*F(N-1)*(DUMMY4**(N-1))/(DUMMY3*
68900              1(1-X66))+(FFD(N)-F(N))*(DUMMY4**N)/(THETR*DUMMY3)
69000   CCC        RES=RES/FOLD(N)
69100              RETURN
69200              END
69300   C
69400   C
69500              SUBROUTINE GOLD(X01,X02,C,XMAX,D,X7,SAI,C65,DUMMY4,ONETH)
69600   C*******************************************************************
69700   C
69800   C              THIS SUBROUTINE SOLVES EQN.4.21 BY GOLDEN SECTION METHOD
69900   C          FOR OPTIMIZATION.
70000   C*******************************************************************
70100              DOUBLE PRECISION X01,X02,C,XMAX,D,X7,SAI,C65,C1,ONETH
70200              DOUBLE PRECISION X3,X4,TEMPX2,Y3,Y4,X1,X2
70300              C1=C/10.0
70400              IF(ABS(X01-X02).LT.C) GO TO 100
70500              X1=X01
70600              X2= X02
70700              X3=X1+0.3820*(X2-X1)
70800              X4=X2-0.3820*(X2-X1)
70900              CALL OBJECT(Y3,X3,D,X7,SAI,C65,DUMMY4,ONETH)
71000              CALL OBJECT(Y4,X4,D,X7,SAI,C65,DUMMY4,ONETH)
71100   50         IF(Y3.GE.Y4) GO TO 30
71200              X1=X3
71300              IF(ABS(X1-X2).LT.C) GO TO 100
71400              TEMPX2=X1+0.3820*(X2-X1)
71500              IF(ABS(TEMPX2-X4).GT. C1) GO TO 40
71600              Y3=Y4
71700              GO TO 45
71800   40         X3=TEMPX2
71900              CALL OBJECT(Y3,X3,D,X7,SAI,C65,DUMMY4,ONETH)
72000   45         X4=X2-0.3820*(X2-X1)
72100              CALL OBJECT(Y4,X4,D,X7,SAI,C65,DUMMY4,ONETH)
72200              GO TO 50
72300
72400   30         X2=X4
72500              IF(ABS(X1-X2).LT.C) GO TO 100
72600              TEMPX2=X2-0.3820*(X2-X1)
72700              IF(ABS(TEMPX2-X3).GT.C1) GO TO 60
72800              Y4=Y3
72900              GO TO 65
73000   60         X4= TEMPX2
73100              CALL OBJECT(Y4,X4,D,X7,SAI,C65,DUMMY4,ONETH)
73200   65         X3=X1+0.3820 *(X2-X1)
73300              CALL OBJECT(Y3,X3,D,X7,SAI,C65,DUMMY4,ONETH)
73400              GO TO 50
73500   100        XMAX=(X1+X2)/2.0
73600              CALL OBJECT(YMAX,XMAX,D,X7,SAI,C65,DUMMY4,ONETH)
73700              YMAXX=YMAX
```

```
73800          RETURN
73900          END
74000   C
74100   C
74200   C
74300   C
74400          SUBROUTINE OBJECT(YOPT,XOPT,D,X7,SAI,C65,DUMMY4,ONETH)
74500   C********************************************************************
74600   C
74700   C                THIS SUBROUTINE REPRESENTS THE OBJECTIVE FUNCTION TO BE
74800   C          OPTIMIZED FOR THE SOLUTION OF EQN. 4.21.
74900   C
75000   C********************************************************************
75100   C
75200          DOUBLE PRECISION YOPT,XOPT,D,X7,SAI,C65,ONETH
75300          YOPT=1.0-XOPT+DLOG(XOPT)+SAI*(1.0-XOPT)*(1.0-XOPT)+C65*X7/
75400         1(D*(DUMMY4**ONETH))
75500          YOPT=-ABS(YOPT)
75600          RETURN
75700          END
```

DATA FOR THE PROBLEM

****

DATA FOR CONSTRAINED OPTIMIZATION_____

NOPT = 11   M = 16   K = 14   ITMAX = 2000   IPRIN = 0

ALPHA =   1.300000   BETA =   0.000100   DELTA =   0.001000

GAMMA = 5   DASHI =   0.900000

INITIAL GUESSES_____

XSTART(1) = 0.40000000E+01   XSTART(2) = 0.99000000E+00
XSTART(3) = 0.16300000E+01   XSTART(4) = 0.99000000E+00
XSTART(5) = 0.99000000E+00   XSTART(6) = 0.64000000E+00
XSTART(7) = 0.99000000E+00   XSTART(8) = 0.14000000E+01
XSTART(9) = 0.30000000E+01   XSTART(10) = 0.60000000E+01
XSTART(11) = 0.24000000E+02   XSTART(

INPUT PARAMETERS FOR LINEAR EQUATIONS AND THE CONVERGENCE CRITERION

MAXIT =   200   NUMSIG = 6   IPT = 1
NUMEQ = 6

EPS = 0.10000000E-07   DELNON = 0.10000000E-06   PREC = 0.50000000E-05
ZETA = 0.10000000E-08

RATE CONSTANTS_____
KP = 0.26500000E+06   KD = 0.23300000E-04   KZERO = 0.00000000E+00
KIP = 0.00000000E+00   KID = 0.00000000E+00   KTC   = 0.72000000E+10
KFT = 0.00000000E+00   KFM = 0.21200000E+02

MONOMER_POLYMER CHARECTARISTICS____

MW   = 0.10400000E+03   DP = 0.10526000E+01   DM = 0.87800000E+00
VMOL = 0.11845000E+03   SAI = 0.25600000E+00

REACTOR CHARECTARISTICS____

THETA = 0.90000000E+03   TEMP = 0.34300000E+03   VW = 0.64940000E+00

OTHER CONSTANTS____

AIOW = 0.15400000E-04   AMZERO = 0.16611000E-12   VD = 0.41888000E-11
VM   = 0.65449800E-19   CE = 0.36720000E+10   AMM = 0.63000000E-02
FEF = 0.10000000E+01   SWC = 0.89000000E-05   YETA = 0.75000000E+01

FEED CONDITIONS____

RADICAL NUMBER DISTRIBUTION____

FFD(1) =   0.000000   FFD(2) =   0.000000
FFD(3) =   0.000000   FFD(4) =   0.000000
FFD(5) =   0.000000   FFD(6) =   0.000000
FFD(7) =   0.000000   FFD(8) =   0.000000

LIVE POLYMER CHAINS_____

FDLIV(0, 1) =   0.0000   FDLIV(0, 2) =   0.0000   FDLIV(0, 3) =   0.0000
FDLIV(0, 4) =   0.0000   FDLIV(0, 5) =   0.0000   FDLIV(0, 6) =   0.0000
FDLIV(0, 7) =   0.0000   FDLIV(0, 8) =   0.0000   FDLIV(0, 9) =   0.0000
FDLIV(0,10) =   0.0000   FDLIV(

FDLIV(1, 1) =   0.0000   FDLIV(1, 2) =   0.0000   FDLIV(1, 3) =   0.0000
FDLIV(1, 4) =   0.0000   FDLIV(1, 5) =   0.0000   FDLIV(1, 6) =   0.0000
FDLIV(1, 7) =   0.0000   FDLIV(1, 8) =   0.0000   FDLIV(1, 9) =   0.0000
FDLIV(1,10) =   0.0000   FDLIV(

FDLIV(2, 1) =   0.0000   FDLIV(2, 2) =   0.0000   FDLIV(2, 3) =   0.0000
FDLIV(2, 4) =   0.0000   FDLIV(2, 5) =   0.0000   FDLIV(2, 6) =   0.0000
FDLIV(2, 7) =   0.0000   FDLIV(2, 8) =   0.0000   FDLIV(2, 9) =   0.0000
FDLIV(2,10) =   0.0000   FDLIV(

FDLIV(3, 1) =   0.0000   FDLIV(3, 2) =   0.0000   FDLIV(3, 3) =   0.0000
FDLIV(3, 4) =   0.0000   FDLIV(3, 5) =   0.0000   FDLIV(3, 6) =   0.0000
FDLIV(3, 7) =   0.0000   FDLIV(3, 8) =   0.0000   FDLIV(3, 9) =   0.0000
FDLIV(3,10) =   0.0000   FDLIV(

FDLIV(4, 1) =   0.0000   FDLIV(4, 2) =   0.0000   FDLIV(4, 3) =   0.0000
FDLIV(4, 4) =   0.0000   FDLIV(4, 5) =   0.0000   FDLIV(4, 6) =   0.0000
FDLIV(4, 7) =   0.0000   FDLIV(4, 8) =   0.0000   FDLIV(4, 9) =   0.0000
FDLIV(4,10) =   0.0000   FDLIV(

FDLIV(5, 1) =   0.0000   FDLIV(5, 2) =   0.0000   FDLIV(5, 3) =   0.0000
FDLIV(5, 4) =   0.0000   FDLIV(5, 5) =   0.0000   FDLIV(5, 6) =   0.0000
FDLIV(5, 7) =   0.0000   FDLIV(5, 8) =   0.0000   FDLIV(5, 9) =   0.0000
FDLIV(5,10) =   0.0000   FDLIV(

DEAD POLYMER CHAINS_____

FDDED(0, 1) =   0.0000   FDDED(0, 2) =   0.0000   FDDED(0, 3) =   0.0000
FDDED(0, 4) =   0.0000   FDDED(0, 5) =   0.0000   FDDED(0, 6) =   0.0000
FDDED(0, 7) =   0.0000   FDDED(0, 8) =   0.0000   FDDED(0, 9) =   0.0000
FDDED(0,10) =   0.0000   FDDED(

FDDED(1, 1) =   0.0000   FDDED(1, 2) =   0.0000   FDDED(1, 3) =   0.0000
FDDED(1, 4) =   0.0000   FDDED(1, 5) =   0.0000   FDDED(1, 6) =   0.0000
FDDED(1, 7) =   0.0000   FDDED(1, 8) =   0.0000   FDDED(1, 9) =   0.0000
FDDED(1,10) =   0.0000   FDDED(

FDDED(2, 1) =   0.0000   FDDED(2, 2) =   0.0000   FDDED(2, 3) =   0.0000
FDDED(2, 4) =   0.0000   FDDED(2, 5) =   0.0000   FDDED(2, 6) =   0.0000
FDDED(2, 7) =   0.0000   FDDED(2, 8) =   0.0000   FDDED(2, 9) =   0.0000
FDDED(2,10) =   0.0000   FDDED(

FDDED(3, 1) =   0.0000   FDDED(3, 2) =   0.0000   FDDED(3, 3) =   0.0000
FDDED(3, 4) =   0.0000   FDDED(3, 5) =   0.0000   FDDED(3, 6) =   0.0000
FDDED(3, 7) =   0.0000   FDDED(3, 8) =   0.0000   FDDED(3, 9) =   0.0000
FDDED(3,10) =   0.0000   FDDED(

FDDED(4, 1) =   0.0000   FDDED(4, 2) =   0.0000   FDDED(4, 3) =   0.0000
FDDED(4, 4) =   0.0000   FDDED(4, 5) =   0.0000   FDDED(4, 6) =   0.0000
FDDED(4, 7) =   0.0000   FDDED(4, 8) =   0.0000   FDDED(4, 9) =   0.0000
FDDED(4,10) =   0.0000   FDDED(

FDDED(5, 1) =   0.0000   FDDED(5, 2) =   0.0000   FDDED(5, 3) =   0.0000

```
FDDED( 5, 4) =    0.0000        FDDED( 5, 5) =   0.0000        FDDED( 5, 6) =   0.0000
FDDED( 5, 7) =    0.0000        FDDED( 5, 8) =   0.0000        FDDED( 5, 9) =   0.0000
FDDED( 5,10) =    0.0000        FDDED(
```

FEED COND. FOR VARIABLES OF INITIAL SET

```
 1      0.4080E+01        2      0.1000E+01        3      0.1920E+01
 4      0.0000E+00        5      0.0000E+00        6      0.0000E+00
 7      0.0000E+00        8      0.0000E+00        9      0.0000E+00
10      0.0000E+00       11      0.0000E+00
```

```
*******************************************************************

          OUTPUT FROM THE PROGRAM_____

*******************************************************************

     (1)   NUMBER OF PARTICLES (/CC EMULSION)____  0.11466412E+16
     (2)   AVERAGE VOLUME OF PARTICLES (CC)_____  0.26767643E-15
     (3)   AVERAGE DIA ;VOLUME BASIS ("A)_____  0.79956124E+03
     (4)   AVERAGE DIA ;NUMBER BASIS ("A)_____  0.71403866E+03

     (5)   CONVERSION OBTAINED_____  0.40438988E+00
     (6)   RATE (MOLES/(LIT.HOUR))_____  0.47828000E+01

     (7)   NUMBER AVERAGE MOLECULAR WEIGHT_____  0.39254813E+06
     (8)   WEIGHT AVERAGE MOLECULAR WEIGHT_____  0.92157680E+06
     (9)   POLYDISPERSITY INDEX_____  0.23476785E+01


*******************************************************************
```

DETAILS OF THE DISTRIBUTIONS FOLLOW_____

****

LIVE POLYMER CHAIN LENGTH DISTRIBUTION

****

NO	MOL. WT.	NUMB.BASIS	WT.BASIS(NORMAL)
0	0.1000E+01	0.1134E-03	0.6037E-07
1	0.7010E+03	0.1116E-03	0.4164E-04
2	0.1401E+04	0.9412E-04	0.7020E-04
3	0.2101E+04	0.7868E-04	0.8801E-04
4	0.2801E+04	0.6548E-04	0.9765E-04
5	0.3501E+04	0.5434E-04	0.1013E-03
6	0.4201E+04	0.4501E-04	0.1007E-03
7	0.4901E+04	0.3722E-04	0.9712E-04
8	0.5601E+04	0.3074E-04	0.9168E-04
9	0.6301E+04	0.2537E-04	0.8511E-04
10	0.7001E+04	0.2092E-04	0.7799E-04
11	0.7701E+04	0.1725E-04	0.7071E-04
12	0.8401E+04	0.1421E-04	0.6356E-04
13	0.9101E+04	0.1171E-04	0.5673E-04
14	0.9801E+04	0.9646E-05	0.5033E-04
15	0.1050E+05	0.7947E-05	0.4443E-04
16	0.1120E+05	0.6549E-05	0.3906E-04
17	0.1190E+05	0.5399E-05	0.3421E-04
18	0.1260E+05	0.4452E-05	0.2987E-04
19	0.1330E+05	0.3673E-05	0.2601E-04
20	0.1400E+05	0.3032E-05	0.2260E-04
21	0.1470E+05	0.2504E-05	0.1960E-04
22	0.1540E+05	0.2069E-05	0.1697E-04
23	0.1610E+05	0.1711E-05	0.1467E-04
24	0.1680E+05	0.1416E-05	0.1266E-04
25	0.1750E+05	0.1172E-05	0.1092E-04
26	0.1820E+05	0.9712E-06	0.9412E-05
27	0.1890E+05	0.8052E-06	0.8103E-05
28	0.1960E+05	0.6680E-06	0.6971E-05
29	0.2030E+05	0.5545E-06	0.5993E-05
30	0.2100E+05	0.4605E-06	0.5148E-05
31	0.2170E+05	0.3826E-06	0.4420E-05
32	0.2240E+05	0.3180E-06	0.3793E-05
33	0.2310E+05	0.2644E-06	0.3252E-05
34	0.2380E+05	0.2199E-06	0.2787E-05
35	0.2450E+05	0.1829E-06	0.2386E-05
36	0.2520E+05	0.1522E-06	0.2042E-05
37	0.2590E+05	0.1266E-06	0.1746E-05
38	0.2660E+05	0.1053E-06	0.1491E-05
39	0.2730E+05	0.8758E-07	0.1273E-05
40	0.2800E+05	0.7281E-07	0.1085E-05
41	0.2870E+05	0.6050E-07	0.9245E-06
42	0.2940E+05	0.5025E-07	0.7865E-06
43	0.3010E+05	0.4170E-07	0.6683E-06
44	0.3080E+05	0.3458E-07	0.5671E-06
45	0.3150E+05	0.2865E-07	0.4805E-06
46	0.3220E+05	0.2371E-07	0.4064E-06
47	0.3290E+05	0.1959E-07	0.3432E-06
48	0.3360E+05	0.1617E-07	0.2893E-06
49	0.3430E+05	0.1332E-07	0.2433E-06
50	0.3500E+05	0.1096E-07	0.2042E-06

```
**************************************************************************

    DEAD POLYMER CHAIN LENGTH DISTRIBUTION

**************************************************************************
```

No	MOL.WT.	NUMB.BASIS	WT.BASIS(NORMAL)
0	0.1000E+01	0.4579E+00	0.4217E-06
1	0.5890E+03	0.7435E-01	0.4060E-04
2	0.1177E+04	0.5342E-01	0.5790E-04
3	0.1765E+04	0.4136E-01	0.6723E-04
4	0.2353E+04	0.3326E-01	0.7206E-04
5	0.2941E+04	0.2739E-01	0.7417E-04
6	0.3529E+04	0.2295E-01	0.7460E-04
7	0.4117E+04	0.1951E-01	0.7396E-04
8	0.4705E+04	0.1677E-01	0.7267E-04
9	0.5293E+04	0.1456E-01	0.7095E-04
10	0.5881E+04	0.1273E-01	0.6896E-04
11	0.6469E+04	0.1121E-01	0.6680E-04
12	0.7057E+04	0.9928E-02	0.6452E-04
13	0.7645E+04	0.8828E-02	0.6215E-04
14	0.8233E+04	0.7877E-02	0.5972E-04
15	0.8821E+04	0.7046E-02	0.5723E-04
16	0.9409E+04	0.6315E-02	0.5471E-04
17	0.9997E+04	0.5666E-02	0.5216E-04
18	0.1059E+05	0.5087E-02	0.4959E-04
19	0.1117E+05	0.4569E-02	0.4701E-04
20	0.1176E+05	0.4102E-02	0.4443E-04
21	0.1235E+05	0.3682E-02	0.4187E-04
22	0.1294E+05	0.3301E-02	0.3933E-04
23	0.1352E+05	0.2957E-02	0.3682E-04
24	0.1411E+05	0.2645E-02	0.3437E-04
25	0.1470E+05	0.2362E-02	0.3198E-04
26	0.1529E+05	0.2106E-02	0.2965E-04
27	0.1588E+05	0.1874E-02	0.2740E-04
28	0.1647E+05	0.1664E-02	0.2523E-04
29	0.1705E+05	0.1475E-02	0.2316E-04
30	0.1764E+05	0.1304E-02	0.2118E-04
31	0.1823E+05	0.1150E-02	0.1930E-04
32	0.1882E+05	0.1011E-02	0.1752E-04
33	0.1941E+05	0.8870E-03	0.1585E-04
34	0.1999E+05	0.7757E-03	0.1428E-04
35	0.2058E+05	0.6762E-03	0.1282E-04
36	0.2117E+05	0.5875E-03	0.1145E-04
37	0.2176E+05	0.5086E-03	0.1019E-04
38	0.2234E+05	0.4385E-03	0.9024E-05
39	0.2293E+05	0.3766E-03	0.7952E-05
40	0.2352E+05	0.3218E-03	0.6971E-05
41	0.2411E+05	0.2737E-03	0.6076E-05
42	0.2470E+05	0.2314E-03	0.5263E-05
43	0.2529E+05	0.1944E-03	0.4527E-05
44	0.2587E+05	0.1622E-03	0.3865E-05
45	0.2646E+05	0.1342E-03	0.3270E-05
46	0.2705E+05	0.1100E-03	0.2739E-05
47	0.2764E+05	0.8908E-04	0.2267E-05
48	0.2823E+05	0.7116E-04	0.1850E-05
49	0.2881E+05	0.5586E-04	0.1482E-05
50	0.2940E+05	0.4287E-04	0.1161E-05

```
****************************************************************

    PARTICLE SIZE DISTRIBUTION [CUMULATIVE]

****************************************************************

    NO          DIA("A)              FRACTION

     0        0.5000E+02           0.0000E+00
     1        0.4553E+03           0.1683E+00
     2        0.5735E+03           0.3083E+00
     3        0.6564E+03           0.4248E+00
     4        0.7225E+03           0.5216E+00
     5        0.7782E+03           0.6022E+00
     6        0.8270E+03           0.6692E+00
     7        0.8706E+03           0.7249E+00
     8        0.9102E+03           0.7713E+00
     9        0.9466E+03           0.8098E+00
    10        0.9805E+03           0.8419E+00
    11        0.1012E+04           0.8685E+00
    12        0.1042E+04           0.8907E+00
    13        0.1070E+04           0.9091E+00
    14        0.1097E+04           0.9244E+00
    15        0.1122E+04           0.9372E+00
    16        0.1147E+04           0.9478E+00
    17        0.1170E+04           0.9566E+00
    18        0.1193E+04           0.9639E+00
    19        0.1214E+04           0.9700E+00
    20        0.1235E+04           0.9751E+00
    21        0.1256E+04           0.9793E+00
    22        0.1275E+04           0.9828E+00
    23        0.1294E+04           0.9857E+00
    24        0.1313E+04           0.9881E+00
    25        0.1331E+04           0.9901E+00
    26        0.1348E+04           0.9918E+00
    27        0.1365E+04           0.9932E+00
    28        0.1382E+04           0.9944E+00
    29        0.1398E+04           0.9953E+00
    30        0.1414E+04           0.9961E+00
    31        0.1430E+04           0.9968E+00
    32        0.1445E+04           0.9974E+00
    33        0.1460E+04           0.9978E+00
    34        0.1474E+04           0.9982E+00
    35        0.1489E+04           0.9985E+00
    36        0.1503E+04           0.9988E+00
    37        0.1516E+04           0.9990E+00
    38        0.1530E+04           0.9992E+00
    39        0.1543E+04           0.9993E+00
    40        0.1556E+04           0.9995E+00
    41        0.1569E+04           0.9996E+00
    42        0.1582E+04           0.9997E+00
    43        0.1594E+04           0.9997E+00
    44        0.1607E+04           0.9998E+00
    45        0.1619E+04           0.9998E+00
    46        0.1631E+04           0.9999E+00
    47        0.1642E+04           0.9999E+00
    48        0.1654E+04           0.1000E+01
    49        0.1665E+04           0.1000E+01
    50        0.1676E+04           0.1000E+01
```

CHE-1979-M-BAL-M A 59503

CHE-1979-M-BAL-M A 59503